Studies on Collision Resolution Algorithms in Communication Systems(Dissertation_全文)

AUTHOR(S):

Oie, Yuuji

STUDIES
ON
COLLISION RESOLUTION ALGORITHMS IN COMMUNICATION SYSTEMS

YUUJI OIE

KYOTO UNIVERSITY
KYOTO, JAPAN
AUGUST 1986

STUDIES

ON

COLLISION RESOLUTION ALGORITHMS IN COMMUNICATION SYSTEMS

YUUJI OIE

KYOTO UNIVERSITY

KYOTO, JAPAN

AUGUST 1986

STUDIES

ON

COLLISION RESOLUTION ALGORITHMS IN COMMUNICATION SYSTEMS

By

YUUJI OIE

Submitted in partial fulfillment of the

requirement for the degree of

DOCTOR OF ENGINEERING

(Applied Mathematics and Physics)

at

KYOTO UNIVERSITY

KYOTO, JAPAN

AUGUST 1986

PREFACE

In a communication system, the basic issue of concern is to efficiently allocate the channel among the users. The solution to this issue has led to various multiple access protocols. Contention-based protocols have been well known to be an effective multiple access protocol in a system consisting of a large number of bursty users. However, in contention-based protocols, as a result of no coordination among the users, it happens that two or more users attempt to transmit their messages on a common channel. This, referred to as 'collision' or 'conflict', causes the performance degradation and, further, instability (or saturation) phenomenon. Tsybakov (in 1978) and Capetanakis (in 1979) independently proposed a novel approach in order to overcome this instability phenomenon. Their protocols are referred to as tree-based protocols (or tree-based collision resolution algorithms (CRAs)) because they use a tree structure in resolving a collision. It is noteworthy that tree-based CRAs stabilize a contention-based communication system without the use of dynamically changing quantities such as an input rate or the number of active users. For this reason, i.e, an excellent property of stability, tree-based CRAs have extensively studied.

The major objective of this dissertation is to develop several new branches of tree-based CRAs and evaluate their performance. The main focus is on both tree algorithms with (control) mini-slots and tree algorithms in a reservation system where a tree algorithm is exploited to schedule an access to the reservation channel. In these two classes of tree algorithms, we further consider two subclasses: free access (FA) and blocked access (BA) subclasses. These subclasses differ by the rules used for transmission of new packets.

First, in chapter2, we propose a BA tree algorithm with mini-slots (BA TA/M), which assumes that binary feedback information (BF) on the state of mini-slot is available, and present an approximate analysis of its average transmission delay. In the next chapter (chapter 3), we analyze the maximum throughput of the BA TA/M as well as that of a BA TA/M-TF, which assumes that ternary feedback information (TF) on the state of mini-slot is available. Furthermore, in chapter 4, we analyze the maximum throughput of two FA TA/Ms, i.e., FA TA/M-BF and FA TA/M-TF. The lower bound of the average transmission delay for these two FA algorithms is explicitly obtained. Through our analysis, in a class of TA/Ms, we show an effect of the feedback information on the state of mini-slot and an effect of how to handle new packets (i.e., blocked access and free access) on the performance.

Second, we consider two reservation algorithms; one exploits a BA-tree algorithm (BA-RTA), and the other exploits an FA tree algorithm (FA-RTA), where Q number of reservation small-slots and L number of data sub-slots are organized into a frame. The maximum throughput of BA-RTA and FA-RTA is analyzed, and an optimum frame configuration is obtained for a given small-slot length. In addition, the performance of BA-RTA and FA-RTA is compared on the basis of these analyses.

Finally, the author would like to hope that the research in this thesis will be helpful for further study in this field.

August, 1986

Yuuji Oie

ACKNOWLEDGEMENTS

I wish to express my sincere appreciation to Professor T. Hasegawa of Kyoto University. His constant encouragement and invaluable comments have greatly helped in accomplishing this research.

I am very grateful to Assistant Professor S. Muro of Kyoto University for his encouragement and guidance throughout the course of this research. I also thank him for his careful reading of the manuscript.

I am also highly indebted to Associated Professor H. Miyahara of Osaka University and Assistant Professor T. Suda of University of California at Irvine for their encouragement and invaluable advice, in particular, concerning the work in chapter 4. I thank Assistant Professor Y. Takahashi, Mr. N.Shima (now in Sumitomo Electric Industries, Ltd.), Mr. H. Shibata (now in IBM Japan), Mr. H.Kawano and the rest of the students in Hasegawa's Laboratory for their encouragement and suggestions.

I wish to thank President K. Matsuda of Sasebo College of Technology and all my colleagues at Department of Electrical Engineering, Sasebo College of Technology for their encouragement.

Finally I would like to express my gratitude to my wife Naoko, who was always willing to hear about my work, for her understanding and help, and to our daughter Yoko for her giving me a pleasant relief. My greatest appreciation goes to our parents for their constant support and encouragement.

CONTENTS

PREFACE iii

ACKNOWLEDGEMENTS v

LIST OF FIGURES ix

LIST OF TABLES xi

CHAPTER 1 INTRODUCTION 1

1.1 Communication System 1

1.2 Multiple Access Protocols 3

1.2.1 Controlled-Access Protocols 3

1.2.2 Contention-Based Protocols 6

1.3 Performance Measures 12

1.4 Previous Studies on Tree Based Algorithms 16

1.4.1 Basic Tree Algorithm 16

1.4.2 Improved Tree Algorithm 19

1.4.3 Dynamic Tree Algorithm 19

1.4.4 0.487 Algorithm 20

1.4.5 CRA with Energy Detectors 20

1.4.6 CRA with Mini-Slots 20

1.4.7 CRA in Reservation Systems and Local Area Networks 21

1.4.8 Other Studies 22

1.5 Overview of the Dissertation 23

CHAPTER 2 TRANSMISSION DELAY ANALYSIS OF ADAPTIVE TREE PROTOCOL . 27

2.1 Introduction 27

2.2 Definitions and Assumptions 31

2.3 Adaptive Tree Protocol 33

2.4 Average Transmission Delay Performance 37

2.5 Numerical and Simulation Results 47

2.6 Conclusions 54

APPENDIX 2-A Derivation of Eq.(2.4) 55

CHAPTER 3 THROUGHPUT ANALYSIS OF BLOCKED ACCESS TREE ALGORITHM WITH MINI-SLOTS 57

3.1 Introduction 57

3.2 Blocked Access Q-ary Tree Algorithms with Mini-Slots . 61

3.3 Analysis of Collision Resolution Time 68

3.4 Throughput Analysis 72

3.5 Numerical and Simulation Results 82

3.6 Concluding Remarks 91

APPENDIX 3-A Proof of Lemma 3.1 92

APPENDIX 3-B Numerical Computations of $g(\alpha)$ 97

APPENDIX 3-C Proof of Corollary 3.1 98

APPENDIX 3-D Throughput Performance of the Basic Tree Algorithm 102

CHAPTER 4 PERFORMANCE EVALUATION OF FREE ACCESS TREE ALGORITHM WITH MINI-SLOTS 103

4.1 Introduction 103

4.2 Free Access Q-ary Tree Algorithms with Mini-Slots . . . 107

4.3 Analysis of Throughput Performance 112

4.4 Evaluation of Average Transmission Delay 123

4.5 Numerical Results 126

4.6 Conclusions 134

APPENDIX 4-A Derivation of Eq.(4.4) 135

APPENDIX 4-B Derivation of Eq.(4.8) 136

APPENDIX 4-C Derivation of Eq.(4.10) 138

APPENDIX 4-D Derivation of Eq.(4.18) 139

APPENDIX 4-E Derivation of Eq.(4.19) 141

CHAPTER 5 THROUGHPUT ANALYSIS OF RESERVATION PROTOCOLS WITH TREE TYPE RESERVATION CHANNEL 142

5.1 Introduction 142

5.2 Reservation Tree Algorithms 146

5.3 Throughput Analysis of BA-RTA and FA-RTA 150

5.4 Concluding Remarks 164

CHAPTER 6 CONCLUDING REMARKS 165

6.1 Summary of this Thesis 165

6.2 Issues for Future Research 170

REFERENCES 172

LIST OF FIGURES

1.1 Classification of multiple access protocols 2

1.2 An address-based partitioning algorithm 10

1.3 Expected drift in an ALOHA system 14

1.4 Research on tree-based algorithms 24

2.1 Slot configuration for the adaptive tree protocol 30

2.2 A tree graph and terminals in a system 32

2.3 Example of transmission process in the adaptive tree protocol (Q=4, K=3) 36

2.4 Queueing model for the adaptive tree algorithm 38

2.5 Initial transmissions in a contention slot 40

2.6 Average transmission delay of the tree protocol (N=1024) . . . 49

2.7 Average transmission delay of the adaptive tree protocol (N=1024, h=0.0) 50

2.8 Average transmission delay of the adaptive tree protocol (N=1024, h=0.005) 52

2.9 Average transmission delay of the adaptive tree protocol (N=1024, h=0.01) 53

3.1 Slot configuration 60

3.2 Example of collision resolution procedure in TA/M-TF(2) with Q=3 65

3.3 Example of collision resolution procedure in TA/M-BF with Q=3 66

3.4 Example of collision resolution procedure in basic TA with Q=3 67

3.5 $g(\alpha)$ versus α 78

3.6 Comparison of $S_{inf}(Q,1)$, $S^*(Q,1)$ and $S_{sup}(Q,1)$ 84

3.7 Comparison of $S_{inf}(Q,2)$, $S^*(Q,2)$ and $S_{sup}(Q,2)$ 85

3.8 Average transmission delay of TA/M-BF (Q=2, 4, 8, 16) 88

3.9 Average transmission delay of TA/M-TF(2) (Q=2, 4, 8, 16) 89

4.1 Slot configuration 106

4.2 Example of transmission process in FA TA/M-TF(2) with Q=3 110

4.3 Example of transmission process in FA TA/M-BF(1) with Q=3 111

4.4 S(Q,m) and its upper bound in FA TA/M-TF(m) 128

4.5 Average transmission delay of FA TA/M-TF(2) (h=0) 131

4.6 Average transmission delay of FA TA/M-BF(1) (h=0) 132

4.7 Average transmission delay of FA TA/M-BF(1) and FA TA/M-TF(2) (h=0.001) 133

5.1 Frame configuration (Q=3, L=2) 145

5.2 Example of transmission process in BA-RTA with Q=3 and L=1 148

5.3 Example of transmission process in FA-RTA with Q=3 and L=1 149

5.4 Throughput ($\frac{k}{B_k}$) vs. multiplicity and ln(Q) in BA-RTA 154

6.1 Maximum throughput of several algorithms analyzed in this thesis 168

LIST OF TABLES

2.1 Numerical and simulation results for average transmission delay in the adaptive tree protocol (Q=8, h=0.0) 51

3.1 $S_{inf}(Q,1)$, $S^*(Q,1)$ and $S_{sup}(Q,1)$ 79

3.2 $S_{inf}(Q,2)$, $S^*(Q,2)$ and $S_{sup}(Q,2)$ 80

3.3 $S_{inf}(Q,3)$, $S^*(Q,3)$ and $S_{sup}(Q,3)$ 81

3.4 $\bar{S}(Q,1)$ in the case of h=0.001 86

3.5 Optimum values of Q and $\bar{S}(Q_{opt},m)$ 87

3.6 $S_{inf}(Q)$, $S^*(Q)$ and $S_{sup}(Q)$ in the basic Q-ary tree algorithm 90

4.1 S(Q,m) in FA Q-ary TA/M-BF(m) 121

4.2 S(Q,m) in FA Q-ary TA/M-TF(m) 122

4.3 $\bar{S}(Q,m)$ in FA TA/M-TF(m) (h=0.001) 129

4.4 Optimum values of Q and $\bar{S}(Q_{opt},m)$ in FA TA/M-TF(m) 130

5.1 $S_{BA-RTA}(Q)$ and $S_{FA-RTA}(Q)$ 161

5.2 Optimum values of L and corresponding values of $\bar{S}_{BA-RTA}(Q,L_{opt})$ 162

5.3 Optimum values of L and corresponding values of $\bar{S}_{FA-RTA}(Q,L_{opt})$ 163

Chapter 1

Introduction

1.1 Communication System

We will consider the situation which consists of geographically distributed stations (users) wishing to communicate over a communication channel. This channel provides the common medium of communication among the stations, and only a single message can be transmitted over the channel at any one time. If two or more messages are simultaneously transmitted on the channel, none of these messages will be correctly received by the destination(s). In such a multiple access environment, the principle problem of concern is how to allocate efficiently the channel among the stations. The solution to this problem yields various multiple access protocols.

A multiple access protocol is regarded as the scheduling algorithm of a channel that is shared by a large number of stations. Multiple access protocols and their performance depend on the environment where they are employed. In a satellite communication channel, the most important characteristic is the inherent long propagation delay of approximately 0.27 sec. In a ground radio channel, the propagation delay is relatively short compared to the transmission time of a packet, and this can be of great advantage in scheduling channel access. Finally, in a local area network, short propagation and high data rate are the main characteristics that are exploited in devising multiple access protocols suitable for this system (see, e.g., [TOBA 80]). Multiple access protocols are evaluated in terms of various performance measures. The most desirable performance characteristics are high throughput (channel utilization) and low message transmission delay.

In this thesis, we will focus our attention on the performance of

- Controlled-Access Protocols
  - Fixed Channel Allocation Protocols
    (FDMA, TDMA)
  - Demand-Adaptive Protocols
    (polling schemes, reservation schemes)
- Contention-Based Protocols
  - ALOHA Type Protocols
    (ALOHA scheme, CSMA, CSMA/CD)
  - Tree-Based Protocols
    (tree protocol,
    interval searching algorithm)

Figure 1.1 Classification of multiple access protocols multiple access protocols in communication systems.

1.2 Multiple Access Protocols

Several authors [TOBA 80, LAM 79, KURO 84] have attempted to classify various multiple access protocols. As previously discussed, the multiple access problem is to grant transmission permissions to a single station with messages to be transmitted over the channel. Multiple access protocols divide broadly into two classes: controlled-access protocol and contention-based protocol (see Fig.1.1).

1.2.1 Controlled-Access Protocols

Controlled-access protocols are characterized by collision free access to the channel. That is, the distributed stations are coordinated in such a way that two or more stations never attempt to transmit messages simultaneously. This coordination is typically achieved by imposing an ordering on the allocation of channel access (transmission) rights to the stations. Controlled-access protocols can be further characterized according to whether or not channel transmission rights are allocated in response to the current transmission requirements of the stations.

1.2.1.1 Fixed channel allocation protocols

Fixed channel allocation protocols allocate the channel to stations in a static manner: i.e., independently of their transmission requirements. As a class of these protocols, frequency division multiple access (FMDA) and time division multiple access (TDMA) are well known. FDMA assigns a fraction of the bandwidth to each station, which sends messages in the allocated subband. On the other hand, TDMA assigns fixed channel time slot to each station, which uses only the allocated slot to send messages. In both these protocols, idle channel (slot) frequently appears at light input rates because fixed channel is assigned to each station, regardless of whether a station has messages. For this reason, these are not well appropriate to the system with a large number of bursty stations. From a performance point of view, it has also been shown that TDMA is superior to FDMA in many cases of practical interest [LAM 77, TOBA 80].

1.2.1.2 Demand-adaptive protocols

As mentioned above, the fixed channel allocation protocols are inefficient at light load. Thus, in order to overcome such inefficiency, demand-adaptive protocols attempt to allocate the channel in a manner more consistent with the immediate transmission requirements of the users. This class contains well-known protocols such as polling schemes and reservation schemes; some of these protocols, e.g., polling schemes [TOBA 76], need a central controller, and others, e.g., the reservation scheme proposed in [CROW 73], operate under a distributed control without a central controller. The token passing scheme developed rapidly in local area networks is viewed as a version of polling schemes [BUX 81].

Polling and Probing

In polling schemes, a central controller polls the users one by one sequentially, and the polled user seizes transmission right. The performance of polling schemes can be expressed in terms of both population size and the propagation delay. As a result of one-by-one polling, the polling schemes perform badly in the case of a large population of users. Recently, polling scheme has been improved in several papers (see, e.g., [HAYE 78, GRAM 80, GUDJ 80, TOWS 84]). We address these studies relevant to this thesis below.

The tree probing was first proposed and analyzed by Hayes [HAYE 78]. This scheme polls groups of users rather than individuals by the use of tree structure; that is, each user corresponds to a leaf in a tree graph, where a user is allocated to a leaf according to his address, and a tree traverse corresponds to an inquiry process. A central controller first asks all the users whether any of them has a message. A user with messages responds affirmatively by putting a noise signal on the channel. If an affirmative answer is received, the group is partitioned into two and the question is repeated to each of the subgroups. The process is repeated until all users having messages are divided into different subgroups. At light loads, a large number of users have no messages, and thus the tree probing is of great advantage. In contrast, if all the users (N) have messages, where N is assumed to be a power of 2, i.e., $N=2^n$, the tree probing needs 2N-1 inquiries per cycle as opposed to only N for the conventional polling. To improve the tree probing, Hayes [HAYE 78] has proposed the adaptive probing such that smaller group of users is initially polled at high traffic loads rather than all of the users [HAYE 78]. The tree probing is also improved in such a way that a user responds affirmatively by putting its address on the channel rather than the noise signal [GUDJ 80]. In this case, a central controller can identify the user sending his affirmative answer if it receives only one answer. Refer to the paper by Towsley et al. [TOWS 84] for other adaptive tree polling algorithms.

Reservation schemes

In this class, a user must acquire (i.e., reserve) the channel over which he will transmit messages. Reservation ALOHA [CROW 73] introduced a frame consisting of several slots. In this protocol, if a user succeeds in transmission without collision in a slot, say i^{th} slot, of a frame, he can acquire the exclusive transmission rights in the i^{th} slot of the following frames as far as he has messages to transmit. This protocol is well suited for the system where a message is divided into several packets of equal length. Another typical reservation scheme is what we call FIFO (first in first out) reservation scheme proposed by Roberts [ROBE 73]. This scheme provides two types of channels; i.e., the request channel used for request packet (or signal) transmission and the data channel for data packet transmission. A user with messages first sends a request packet in request channel, and after succeeding in transmitting the request packet correctly, he transmits messages in the data channel on the basis of FIFO discipline (see [ROBE 73]).

In reservation schemes, a user can transmit his messages without collision after acquiring (or reserving) transmission rights in the data channel. As an access scheme used for reserving the data channel, controlled-access protocols or contention-based protocols are employed; i.e., access to the request channel yields another multiple access problem. Refer to the papers such as [JACO 78, KLEI 80, LEE 83, RETN 80] for other reservation schemes.

1.2.2. Contention-Based Protocols

The second broad class of multiple access protocols, known as contention-based protocols, were developed to overcome the inefficiency due to fixed channel assignment and the control overhead due to demand-adaptive protocols.

1.2.2.1 ALOHA Type Protocols

Abramson [ABRA 70] devised the first contention-based protocol, which is known as pure ALOHA. The basic concept of the ALOHA system is simple; a user simply transmits a message when it arrives. This protocol requires no coordination among the distributed users. If two or more users simultaneously transmit messages at the same time, their messages interfere (or collide) and thus require retransmission.

As previously mentioned, a multiple access protocol strongly depends on the environment in which it is employed. In a ground radio environment, the propagation delay time is relatively short compared to packet transmission time. Thus, by monitoring the channel, the stations can take an advantage of the current state of channel in scheduling transmissions. This class is known as the carrier sense multiple access (CSMA) protocols, two of which, i.e., non-persistent and p-persistent CSMA, were analyzed by Kleinrock and Tobagi [KLEI 75]. Furthermore, in a local area network, a propagation delay is short enough for a user to detect whether or not his transmission collides prior to the end of transmission. Owing to this capability to detect a collision promptly, if a collision arises, all the stations can abort their transmissions and then retransmit later. This mechanism helps to reduce a waste of the channel due to collided transmissions. This sort of protocols are known as the CSMA with collision detection (CSMA/CD), and its prominent example is the Ethernet [METC 76].

As noted above, in contention-based protocols, as a result of no coordination among distributed users, two or more users may transmit messages at the same time; i.e., a collision is inevitable. Thus, the main issue of concern is how to efficiently resolve a collision.

The earliest approach to this problem is uniform retransmission randomization scheme [KLEI 75]. In this scheme, a user defers a retransmission for a random amount of time whose maximum value is fixed. The probabilistic element in the retransmission scheme helps collided users to eventually retransmit at different times. This scheme restricted the maximum value of retransmission interval to fixed one. On the other hand, the following two retransmission (backoff) algorithms vary the maximum value of retransmission interval in response to the number of collisions experienced by the packet [TROP 83]: the binary exponential backoff algorithm, which is employed in the Ethernet, and the linear incremental backoff algorithm. In addition, another kind of contention-based protocol, the URN scheme, was proposed by Kleinrock and Yemini [KLEI 78]; when the number of active users in a network is given, it grants transmission permissions to a subset of users, whose population size is determined in such a way as to maximize the probability that the subset contains exactly one active user (i.e., user having messages to be transmitted).

These algorithms have common undesirable property; that is, the inherent instability phenomenon, which has been pointed out in several protocols: in the slotted ALOHA [FERG 75, FAYO 77], the CSMA [TOBA 77], the CSMA/CD [ROSE 84] and the URN scheme [MITT 81]. Several papers have proposed control policies that stabilize these schemes [LAM 75, FAYO 77, TOBA 77, MITT 81].

1.2.2.2 Tree-Based Protocols

Another solution to instability problem leads to extensive studies on tree-based protocols. Tree-based protocols refer to how to partition the active users into a set of enabled users (users with transmission rights) and a set of disenabled users (users without transmission rights) rather than how to schedule the time instant of a retransmission. Thus they can be regarded as partitioning algorithms.

If none of the enabled users are active, then the channel remains unused and a new partitioning of the users is eventually determined. If exactly one enabled user is active, a new partitioning is then determined. Finally, if two or more active users are in the enabled set, then their messages collide. If users detect collided transmissions, then the enabled set is further partitioned in order to isolate a single active user in the enabled set. Numerous mechanisms have been proposed to determine the partitioning of users in tree-based protocols. We will mention three kinds of tree-based protocols below: address-based, probabilistic and time-based partitioning algorithms.

Address-based partitioning algorithm

This class of algorithms was first proposed by Tsybakov [TSYB 78] and Capetanakis [CAPE 79a,b]. These algorithms permit a subset of users to transmit their packets in the current slot based on their addresses. We will refer to the algorithm proposed by Capetanakis (i.e., binary tree algorithm) and its extended version (i.e., Q-ary tree algorithm presented by Mathys [MATH 85]) as the basic tree algorithm.

An address-based Q-ary basic tree algorithm operates as follows [CAPE 79a, MATH 85]. We can think of each of the users as a leaf in a Q-ary tree graph, as shown in Fig.1.2. If a population size N is given by Q^n, each user has a Q-ary address with n figures, and through these n figures, a packet is successfully transmitted after at most n number of retransmissions. We assume that channel time is slotted and users with a packet initially attempt to send their packets in the current slot. If a collision arises, the set of users involved in the collision is partitioned into Q subtrees according to their addresses. Only users in one of Q subtrees of the Q-ary tree are allowed to transmit in the subsequent time slot. If further collisions occur, the enabled set is continually divided into Q subtrees until the enabled set eventually has only one ready user.

Figure 1.2 shows an example of a collision resolution process of a binary tree algorithm. This system consists of 8 ($=2^3$) users with binary addresses. In slot 1, users B, C, G and H transmitted their packets, resulting in a collision. Then, transmission rights are given to users B and C involved in the left subtree. Since this subtree has more than one

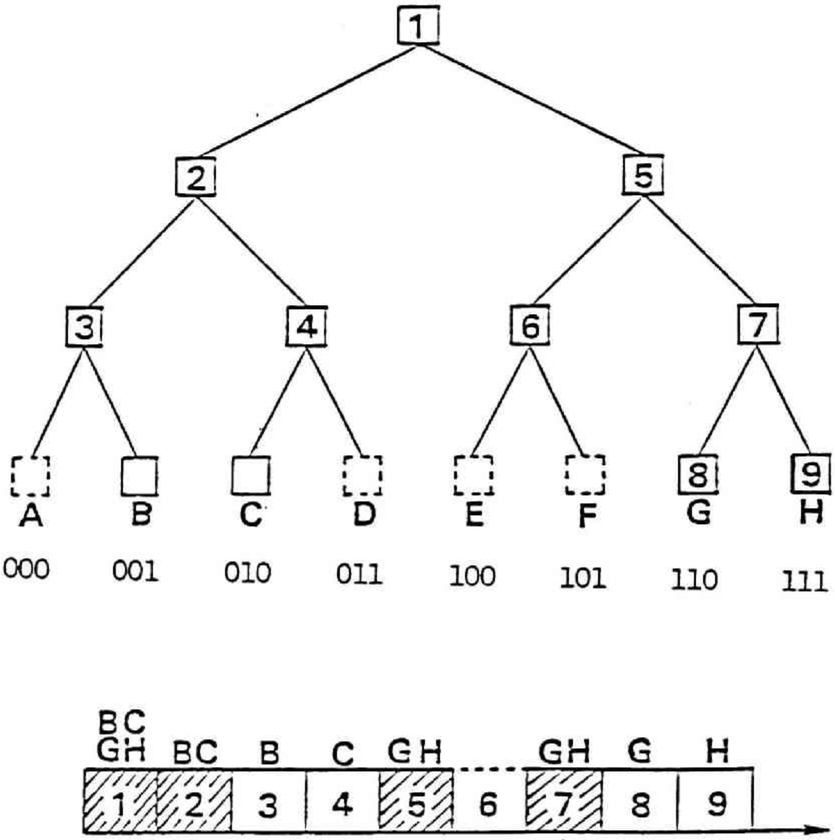


Figure 1.2 An address-based partitioning algorithm

(the initial collision in slot 1 is due to user B, C, G and H) active user, it is further divided into two subtrees. These subtrees has only one active user, and a collision of the subtree rooted at node 2 is resolved. On the other hand, users G and H are forced to wait until a collision of the left subtree is resolved. A collision due to users G and H is resolved in a similar way.

Probabilistic partitioning algorithm

In this class of algorithms, random coin tosses are exploied instead of users' addresses in partitioning a set of users. This class of algorithms can be also viewed as the limiting case of an infinite population size in address-based tree algorithms.

Time-based partitioning algorithm

This algorithm partitions a set of packets according to their arrival times. It has first been proposed by Gallager [GALL 78], Tsybakov [TSYB 80d] and Ruget [RUGE 81], independently. Window algorithms can be regarded as being of this class (see, e.g., [TOWS 82, KURO 84]). We can think these algorithms as a modified version of the tree algorithm. These basically operates as follows.

It is assumed that users can recognize that a slot is empty, successful or collided. The algorithm begins by granting transmission permissions to all the packets that generated during a time interval of length (I). When a collision occurs, the interval is divided and the left half (of length I/2) is first enabled. Whenever a collision arises, the interval is further divided into halves and the left half is first enabled. When an empty slot occurs, the right half is immediately partitioned and then its left half is enabled. Finally, when a packet is correctly transmitted, the right half is enabled. This process continues until two consecutive successful transmissions are detected.

1.3 Performance Measures

As basic performance measures, we often use throughput (channel throughput rate), transmission delay, stability, robustness, etc. In this section, we assume that each packet is transmitted in one time unit (slot) of duration equal to transmission time of packet, and a system consists of an infinite number of users.

Average transmission delay

The performance measures of practical significance are, first of all, an average transmission delay and throughput. The transmission delay usually represents the time elapsed from the instant that a packet arrives to the instant that the packet is correctly received in the absence of collision.

Throughput

Let us define throughput to be the probability that a packet is correctly transmitted per packet transmission time. Capacity (channel capacity) is often used as the maximum possible throughput [KLEI 75].

We consider the throughput of slotted ALOHA [KLEI 73]. Now, let S denote the throughput and G the offered traffic, which contains both initial transmissions due to newly arriving packets and retransmissions due to previously collided packets. Assuming that new packets arrive according to Poisson distribution, we have the following equation (see [KLEI 73]).

$$S=Ge^{-G}.$$

It is easily obtained that the throughput achieves the maximum value (i.e., capacity) of $1/e=0.368$ at $G=1$ from the above equation. Note that S decreases as G becomes larger than 1, and approaches 0 as G does infinity.

Stability

Kleinrock and Lam have defined a stable and an unstable channel as follows [KLEI 75]:

> "In a stable channel, equilibrium throughput-delay results are achievable over an infinite time horizon. In an unstable channel, such channel performance is achievable only for some finite time period before the channel goes into saturation."

Channel saturation in the above sentence refers to the phenomenon such that the channel throughput becomes zero as a result of the increase of collisions and retransmissions.

Now, we will see the instability of the slotted ALOHA with uniform retransmission randomization scheme [CARL 75, KLEI 75]. In this retransmission scheme, each backlogged packet, i.e., the packet which experienced collisions, is retransmitted in the current slot with probability p. Let n be the number of backlogged packets, and $S_n(p)$ be throughput conditional on n. $S_n(p)$ is given by (see [KLEI 75])

$$S_n(p)=\lambda e^{-\lambda}(1-p)^n+e^{-\lambda}n(1-p)^{n-1}.$$

We note that

$$\lim_{n\longrightarrow\infty} S_n(p)=0.$$

Let, further, $D_n(p)$ denote the expected drift from n (see [CARL 75]); we have

$$D_n(p)= \lambda-S_n(p).$$

The equation $D_n(p)=0$ is interpreted as the equilibrium condition that an input rate equals an output rate (i.e., throughput). It should be noted that the equation $D_n(p)=0$ has two solutions for any fixed value of λ, i.e., two equilibrium states.

Figure 1.3 illustrates $D_n(p)$ for some values of p and λ, and n_A and n_B represent equilibrium states. In the case that the the number of backlogged packets satisfies that $D_n(p)>0$, an input rate is larger than the capacity at this time, so that the number of backlogged packets drifts in the direction to becoming larger. Conversely, when the number

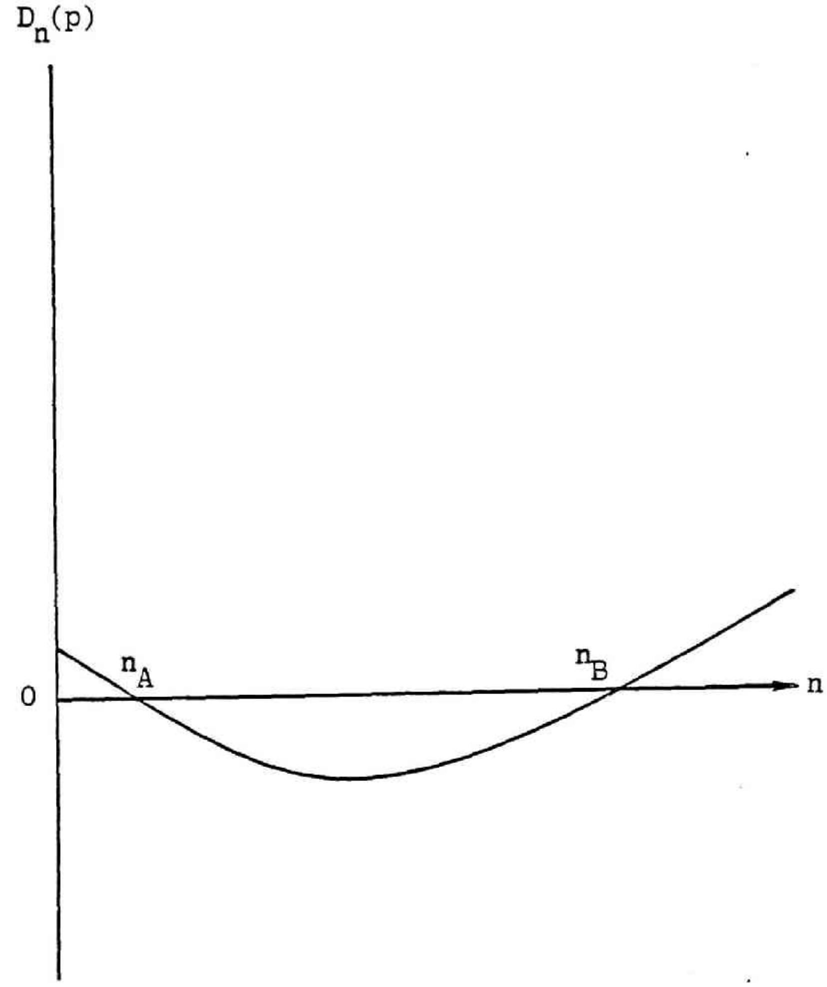


Figure 1.3 Expected drift in an ALOHA system

of the backlogged packets satisfies that $D_n(p) < 0$, it becomes smaller and smaller. As a result, when the number of backlogged packets is less than n_B, it approaches n_A; and when the number of backlogged packets is larger than n_B, it drifts in the direction of becoming larger and larger; at last, no packet will be transmitted correctly independently of traffic intensity of new packets. This phenomenon is referred to as a channel saturation as stated above. Consequently, n_A is a stable equilibrium state, but the other, n_B, is unstable one. Stochastic fluctuations may cause such saturation phenomenon.

Several papers (see, e.g., [METC 73, CARL 75, KLEI 75, FAYO 77]) have shown that the slotted ALOHA exhibits such unstable behavior (or bistable behavior) on the basis of the assumptions of an infinite as well as finite population. In particular, Fayole et al. [FAYO 77] proposed the stable 'optimal' ALOHA scheme. This scheme chose p as a function of n and λ, say p^*; p^* is given by $p^*=(1-\lambda)/(n-\lambda)$. In this case,

$$\lim_{n \to \infty} S_n(p^*)=\frac{1}{e}.$$

Namely, this scheme attains the maximum throughput 1/e at all times, and thus exhibits stable behavior. However, this algorithm is based on the assumption that n and are given or can be estimated, and thus is not useful in practice. Another solution to the instability issue leads to various studies regarding tree algorithms which do not require such dynamically changing quantities (n and λ) or adaptive control as we shall see later.

Robustness

It is desirable that a protocol is insensitive to channel error. Several papers discussed a robustness to channel error of tree type collision resolution algorithms [MASS 80, GEOR 85].

1.4 Previous Studies on Tree Based Algorithms

In the following, we will often use the term "algorithm" or "collision resolution algorithm (CRA)" instead of protocol; all of these terms are equivalent in contention-based protocols.

In the last decade, a broad spectrum of studies on tree-based protocols have been carried out. It is worth noting that tree protocols offer stable behavior without such dynamic controls as proposed in [LAM 75, FAYO 77]. For this desirable property, various tree-based protocols have been proposed and analyzed (see [MURO 86a]). Recently, a special issue on random-access communications was published in IEEE Transaction on Information Theory [IEEE 85]. Interestingly, almost all the papers involved in this issue treated CRAs which will be described below. This indicates how the studies of CRAs are active. Therefore, in this section, we will particularly focus our attention to tree-based protocols and related studies. We will give a brief description of mechanisms of these algorithms, their performance and related studies.

1.4.1 Basic Tree Algorithm

The Q-ary basic tree algorithm was analyzed by Mathys [MATH 85]. Its original version (i.e., the binary tree algorithm) was proposed by Capetanakis [CAPE 79a]. In the basic tree algorithm, there are two types: an address-based and a probabilistic basic tree algorithm. The mechanism has been already described in section 1.2.2.2.

Blocked Access and Free Access

Basic tree algorithms broadly divide into two classes depending on how new packets are handled: free access and blocked access tree algorithms. Free access tree algorithms do not distinguish between new and collided packets. User attempts to transmit a new packet immediately in the slot succeeding its generation. Blocked access tree algorithms force new packets to join a queue at a user and to wait until all the outstanding collided packets have been removed from a system. Transmission of new packets starts only after all the outstanding collisions have been resolved. Note that, in blocked access tree algorithms, each user keeps monitoring the channel even when he has no packet to send. This is necessary to keep track of the (system-wide) outstanding collided packets and to determine the time to start transmission of a new packet. A free access tree algorithm is base on a probabilistic partitioning algorithm, and on the other hand, a blocked access tree algorithm can be implemented by means of either address-based algorithm or probabilistic partitioning algorithm.

Stability

Here, for example, we give a statement for the stability of blocked access binary tree algorithm. Assuming that the propagation delay is equal to zero, a sequence of the number of packets transmitted in the first slot of collision resolution intervals (CRIs) forms a Markov chain. Let X_i be the random variable representing the number of such packets in the i^{th} CRI. Then, $E[X_i-X_{i+1}|X_i=n]$ can be regarded as the expected drift from n number of backlogged packets in the first slot of the i^{th} CRI. Denoting by T_n the conditional average collision resolution time (CRT), $E[CRT|X_i=n]$, we have

$$E[X_i-X_{i+1}|X_i=n]=i-T_n\lambda.$$

Massey obtained the following tight bounds of T_n as a linear function of n [MASS 80]:

$$2.8810n-1\leq T_n\leq 2.8867n-1 \quad (n\geq 4).$$

From this equation, when an input process is a Poisson distribution whose mean value λ is less than 1/2.8867=0.3463,

$$E[X_i - X_{i+1} | X_i = n] < 0 \quad (n \geq 4).$$

This inequation indicates that, if the number of backlogged packets was equal to or larger than 4, the number of backlogged packets will become smaller regardless of how many packets were backlogged. Namely, in contrast to the slotted ALOHA system (see in section 1.3), this system does not suffer from channel saturation.

Furthermore, if an input rate is less than 0.3463, we obtain the following properties:

$$|E[X_{i+1} - X_i | X_i = n]| < \infty ,$$

$$\lim_{i \longrightarrow \infty} \sup E[X_{i+1} - X_i | X_i = n] < 0.$$

Therefore, from a theorem of Pakes [PAKE 69], this Markov chain is ergodic and thus this system is stable under the condition, $\lambda < 0.3463$. In addition, its value is the stable maximum throughput of the blocked access binary tree algorithm.

As mentioned above, the blocked access tree algorithm exhibits stable behavior. The above tight bounds on T_n imply that this algorithm attains a throughput of 0.3463 even when n approaches infinity. The stability of a free access tree algorithm can be shown in a different way [MATH 85].

Throughput Performance

Throughput analysis has been done and the maximum throughput of both free access and blocked access tree algorithms has been obtained. For each algorithm, ternary division of a tree (Q=3) gives the best maximum throughput, and its value is 0.4016 [MATH 85] in the free access case and is 0.3622 [MATH 85, MURO 85] in the blocked access case. That is, a free access tree algorithm outperforms a blocked access tree algorithm. This is because, in a free access algorithm, new packets can take an advantage of slots that would be left unused in a blocked access algorithm because, in a blocked access tree algorithm, new packets are denied to access to the channel until all the outstanding collisions have been resolved.

Several improvements have been considered on the basic tree algorithms described below.

1.4.2 Improved Tree Algorithm

In the basic tree algorithm, if a collision arises and (Q-1) empty slots succeed, the following slot will suffer from collision for sure. In the improved tree algorithm, in such a case, all the packets involved in the most recent collision are immediately partitioned into Q subgroups before retransmission. In this manner, the improved tree algorithm eliminates this kind of predictable collisions, which would arise in the basic tree algorithm.

The throughput analysis was carried out for both of blocked access [TSYB 78, MATH 85] and free access algorithms [TSYB 80c, MATH 85]. In both of these algorithms, binary division of a tree (Q=2) gives the best throughput: 0.3754 in blocked access algorithm and 0.4070 in free access one. Mathys also analyzed the anti-symmetrical case [MATH 85].

1.4.3 Dynamic Tree Algorithm

Capetanakis [CAPE 79a,b] and Massey [MASS 80] applied a dynamic algorithm to the blocked access basic and improved tree algorithms. In both blocked access algorithms, as previously mentioned, all the newly arriving packets during the i^{th} CRI are transmitted in the first slot of the $i+1^{st}$ CRI. On the other hand, the dynamic tree algorithm permits only the packets arriving during the i^{th} arrival epoch, which is the time interval (ih, ih+h], to be transmitted in the first slot of the $i+1^{st}$ CRI. By optimizing the values of h, they improved the maximum throughput of the basic tree algorithm from 0.346 to 0.429 (which was attained by

$\lambda h=1.147$), and that of the improved tree algorithm from 0.375 to 0.462 (which was attained by $\lambda h=1.251$).

1.4.4 0.487 Algorithm

This class of algorithms are time-based partitioning algorithms, and are also referred to as part-and-try [TSYB 80d] or an interval searching algorithm. As previously mentioned, this algorithm terminates collision resolution procedure if two consecutive successful slots ensure, and the remaining packets will be served together with newly arriving packets. The algorithm previously described in section 1.2.2 attains a maximum throughput of 0.48711 [GALL 78, TSYB 80d, RUGE 81]. Mosely refined this algorithm by optimizing at every step the length of the enabled interval to obtain a maximum throughput of 0.48776 [MOSE 85]. This algorithm is thus far the best algorithm among the algorithms employing the 0,1,e-feedback information (see, [BERG 84]); i.e., users can recognize that a slot is empty, successful or collided.

1.4.5 CRA with Energy Detectors

This algorithm has been analyzed by Tsybakov et al. [TSYB 80b] and Geordiadis et al. [GEOR 83]. It assumes that users can recognize the multiplicity of collision, i.e., the number of packets involved in a collision. This information can be made available to users by detecting signal power level on the channel. By assuming this additional feedback information, the maximum throughput was shown to increase up to 0.53237.

1.4.6 CRA with Mini Slots

There is another class of CRAs to further improve the throughput performance of tree algorithms; Q-ary tree algorithms with mini-slots. In this class of algorithms, Q number of mini-slots are provided within a (large) slot to allow users to acquire additional information on the state of the packet transmission. Data sub-slot length is equal to packet transmission time. When a user sends a packet (using a data sub-slot in a large slot), he also sends a signal in a mini-slot randomly chosen. In case of a collision, the current enable set of users are divided into Q number of subtrees, each corresponding to a group of users who have chosen the same mini-slot. A great advantage of this class of algorithms over the basic one is that users can tell which subtrees are active (i.e., have active users in it) and thus, slots will be only assigned to active subtrees. Note that, in basic tree algorithms, since users do not know which subtrees are active (or non-active), a slot will be also assigned to non-active subtrees.

This class of algorithms have studied in such papers as [MERA83, BERG 83, HUAN 83, HUAN 85].

1.4.7 CRA in Reservation Systems and Local Area Networks

As previously stated in section 1.2.1.2, reservation scheme is a member of controlled-access protocols. As a way to reserve a data channel, fixed channel allocation or contention-based protocols are exploited. The reservation ALOHA [ROBE 73] and the FIFO reservation scheme [ROBE 73] are well known as contention-based reservation scheme. These schemes are efficient in a system containing a large number of users, but, as the slotted ALOHA, the instability phenomenon was pointed out in these schemes [SZPA 83, TASA 84]. Thus, several papers have attempted to apply a tree algorithm to reservation systems (see, e.g., [TSYB 80a, LEE 83]).

In local area networks, tree-based CRAs are also employed in several papers (see, e.g., [MERA 85, MURO 86b]) in order to overcome the instability issue underlying the well-known CSMA/CD [MEDI 83].

1.4.8 Other Studies

First, the study of practical interest is associated with what we call limited sensing algorithm [GERG 85, HUMB 86]. This kind of algorithms do not require each user to monitor the channel at all times in contrast to the basic blocked access tree algorithm [CAPE 79a], and this property is very desirable to implement. A free access algorithm is of this class. In a free access environment, the maximum attainable throughput was obtained [HUMB 86].

Next, Berger et al. [BERG 84] applied the group testing technique [SOBE 59], which was a branch of applied statistics, to collision resolution procedure in a random access environment. The group testing algorithm grants transmission permissions to a set of users at each step based on the optimal criterion that minimizes the average number of group tests required to identify all the packets to be transmitted. The group testing algorithm is similar to adaptive polling [HAYE 78] and dynamic tree algorithm [CAPE 79, MASS 81] in testing a group of users, whereas it does not depend on a tree structure. In [BERG 84], assuming 0,1,e-feedback, it was shown to be optimum to test all the users one by one when the probability that each user has packets is larger than $1/\sqrt{2}$. The works regarding this algorithm have been carried out in several papers (see, e.g., [SESH 85, WOLF 85]).

Finally, as noted in section 1.4.4, Mosely and Humblet obtained the maximum throughput of 0.48776 assuming 0,1,e-feedback information. Then, how much is an upper bound on the maximum attainable throughput on the basis of the same assumption? Pippenger first took an approach to this problem, and estimated the upper bound as 0.744 [PIPP 81]. However, its upper bound was not so tight, and thus several papers subsequently attacked the same issue [MOLL 82, CRUZ 82, THOM 84]. Recently, Panwar et al. [PANW 85] sharpened the upper bound to 0.5, which is the tightest one so far.

1.5 Overview of the Dissertation

As mentioned in section 1.4, tree-based protocols have constituted an important class of multiple access protocols. The maximum throughput of the simplest binary tree algorithm, i.e., Capetanakis' static algorithm, is less than that of the slotted ALOHA, but is stable throughput. Thereafter, the works of Capetanakis [CAPE 79a,b], Tsybakov [TSYB 78] and Massey [MASS 80] have followed by various studies, and the performance has been improved. The major objective of this dissertation is to improve the performance of the tree algorithm and analyze the performance of several tree-based algorithms.

Figure 1.4* illustrates the previous research on tree-based algorithms and the works treated in this thesis. Each chapter of this dissertation is as follows.

In chapter 2, a blocked access tree protocol with control mini-slots is proposed, and it is referred to as the adaptive tree protocol. In this protocol, each user is assumed to distinguish between an empty mini-slot and a busy mini-slot; mini-slots provide binary feedback information on channel state. The information provided by mini-slots helps to completely eliminate empty slots in a collision resolution interval (CRI). In a system where a round-trip delay is too large to be negligible, the average transmission delay is analyzed and simulation results are also shown

* In Fig. 1.4, inherent feedback information refers to information that users can obtain without any special assumption; i.e., without energy detectors or mini-slots. The basic TAs requires only binary inherent feedback information; in these algorithms, it is assumed that users can distinguish between a collided and a non-collided slot. On the other hand, the improved tree algorithms and the 0.487 algorithm require inherent ternary feedback information: 0,1,e-feedback information.

- direct access
  - TA with inherent feedback information
    - basic TA
      - blocked access: binary [CAPE 79a, MASS 80, HOFR 84]
        Q-ary [MATH 85, MURO 85]
        optimum degree=3
        maximum throughput in this case=0.3662
      - free access : Q-ary [MATH 85]
        optimum degree=3
        maximum throughput in this case=0.4016
    - improved TA
      - blocked access: binary [TSYB 78, MASS 80]
        Q-ary [MATH 85]
        optimum degree=2
        maximum throughput in this case=0.3754
      - free access : binary [TSYB 80c]
        Q-ary [MATH 85]
        optimum degree=3
        maximum throughput in this case=0.4070
    - dynamic TA
      - dynamic basic TA [CAPE 79, MASS 80]
        maximum throughput =0.429
      - dynamic improved TA [MASS 80]
        maximum thorughput =0.462
    - 0.487 algorithm [GALL 78, TSYB 80d, MOSE 85, HUAN 85]
      best algorithm was proposed by Mosely et al. [MOSE 85]
      maximum throughput in [MOSE 85]
      =0.48776
  - TA with noninherent (or additional) feedback information
    - TA with energy detectors [TSYB 80b, GEOR 83]
      maximum throughput =0.53237
    - TA with control mini-slots
      - blocked access :[MERA 83, BERG 83, HUAN 85, OIE 84, OIE 85a, MURO 85, SZPA 85]
        (Chapter 2, Chapter 3)
      - free access :[OIE 87]
        (Chapter 4)
- reservation access
  - without reservation channel: [TSYB 80a]
  - with reservation channel **(Chapter 5)**
    - blocked access: [LEE 83, OIE 85b, OIE 86]
    - free access : [OIE 86]

Figure 1.4 Research on tree-based algorithms discussing the performance of the algorithm.

In chapter 3, we consider a blocked access tree algorithm with Q number of mini-slots, which inform users of the channel state. We study two cases regarding the feedback information available in mini-slots; binary feedback (it was also treated in chapter 2) and ternary feedback information. The throughput characteristic of the algorithm is analyzed, and an optimum value of Q is obtained to maximize throughput for a given mini-slot length. On the basis of the throughput analysis and simulation results, a comparison of the performance of above two algorithms is shown.

Chapter 4 provides performance evaluation of two kinds of free access tree algorithms with Q number of mini-slots per data slot. One is that binary feedback information is available in mini-slots, and the other is that ternary feedback information is available. The maximum throughputs of these algorithms are analytically obtained for a given value of Q, and it is also shown that, in these algorithms, the highest throughput 0.56714 is achieved in the limiting case where Q approaches infinity and the length of mini-slot goes to zero. The throughput 0.56714 is equal to the achievable maximum throughput first obtained by Humblet in the context of free and direct channel access. In addition, an explicit expression of the lower bound on the average transmission delay is derived. It is worth noting that the derived lower bound is the lower bound on the average delay of all free access algorithms.

In chapter 5, we consider two reservation protocols (B-RTA and F-RTA); as a scheduling algorithm of an access to reservation channel, one exploits a blocked access tree algorithm, and the other a free access tree algorithm. It is assumed that a frame consists of Q number of small-slots (reservation channel) and L data sub-slots. We obtain the maximum throughput of both algorithms. Through our analyses, an optimum frame configuration is obtained for a given small-slot length. In the special case that a frame consists of Q number of small-slots and one data sub-slot, Lee and Mark [LEE 83] analyzed the average transmission delay of a B-RTA and showed through numerical results for the average transmission delay that Q=3 gives an optimum performance. Our result agrees with their result. On the basis of throughput analyses, we compare the performance of both algorithms.

In chapter 6, some concluding remarks and suggestions for future research are given.

The results discussed in chapter 2 is mainly taken from [OIE 84], chapter 3 from [OIE 85a, MURO 85], chapter 4 from [OIE 87], and chapter 5 from [OIE 85b, OIE 86].

Chapter 2

Transmission Delay Analysis of Adaptive Tree Protocol

2.1 Introduction

In multiple access environments, various protocols have been proposed and analyzed (see chapter 1). Contention-based protocols have been devised in an attempt to enable a large number of users to efficiently share a single communication channel among them. Yet these protocols suffer collision due to simultaneous transmissions of two or more packets. A collision reduces system performance, so that the main issue of concern is how to efficiently resolve a collision. Among various collision resolution algorithms (CRAs), the tree type CRAs are excellent algorithms in view of their channel stability.

The tree type CRAs were proposed and have been investigated by Capetanakis [CAPE 79a,b], Massey [MASS 80], Tsybakov and Mikailov [TSYB 78] and so on. Capetanakis proposed tree protocol with a throughput of 0.346; we will refer this protocol as the basic tree protocol (see section 1.2.2.2 for the mechanism of the protocol). In addition he also proposed dynamic tree protocol with a throughput of 0.429. Subsequently, Massey improved the maximum throughput of the basic tree protocol to 0.375 in a static case and to 0.462 by a dynamic algorithm.

Performance of the tree protocol is characterized by the time required to resolve a collision; the time comprises three kinds of slots: empty, successful and collided slots. In the basic Q-ary tree protocol, whenever a collision arises, the packets involved in the collision are partitioned into Q subgroups, each of which is assigned a slot for retransmission. A subgroup with no packet leads to an empty slot; a subgroup with exactly one packet yields a successful slot; a subgroup with two or more packets results in a collided slot. It is obvious that empty and collided slots contribute to a degradation of performance.

In this chapter, we attempt to improve the basic tree protocol in such a way to eliminate the empty slots during a collision resolution interval. We therefore introduce mini-slots that, in the event of collision, provide information about which subgroup is empty (i.e., which subgroup contains no packet) before retransmissions of the packets involved in the collision. Thus, a (large) slot is assumed to consist of Q number of mini-slots and a data sub-slot (see Fig.2.1). We refer to the protocol considered in this chapter as an adaptive tree protocol.

Here, we will describe an outline of the adaptive tree protocol. When a user sends a packet in a data sub-slot, he simultaneously sends a signal in a mini-slot according to his address. In case of a collision, the current enable set of users (to transmit) are divided into Q subtrees, each corresponding to a group of users who have chosen the same mini-slot. A great advantage of this protocol is that users can recognize which subgroup is active (i.e., has active users in it) and thus slots will be assigned to only active subgroups. Note that, in the basic tree protocol, since users do not know which subgroups are active (or non-active), a slot will be also assigned to a non-active subgroup. Our protocol is adaptive in the sense that the slot assignment dynamically varies in response to the number of active subgroups.

In the above stated tree protocols, there are two possible partitioning methods; they are on the basis of user's address or random coin toss. The former has a good property that it surely resolves a collision with a finite number of steps. The former is treated in this chapter, and the latter will be treated in the following chapter (chapter 3). We will propose the adaptive tree protocol and approximately analyze its average transmission delay in a system with large propagation delay. Especially, we will consider a satellite communication system. As for the maximum throughput, it will be analyzed in chapter 3. Through our analysis, it will be shown that the improvement on the throughput vs. average transmission delay performance and its insensitivity to a population size. Section 2.2 gives definitions and assumptions and section 2.3 gives an exact description of the adaptive tree protocol. In section 2.4, the average transmission delay for the adaptive tree protocol is approximately analyzed. In section 2.5, numerical and simulation results are shown.

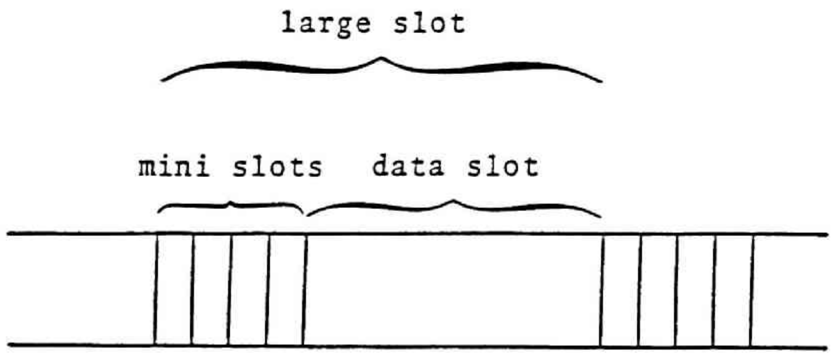


Figure 2.1 Slot configuration for the adaptive tree protocol

2.2. Definitions and Assumptions

We shall give several definitions of the tree graph according to [CAPE 79a] (see Fig.2.2).

depth of a node: the number of branches between the node and the root node, where the root node is at depth zero.

depth of a tree graph: the depth of the leaf (bottom node) of a tree.

degree Q of node: the number of branches that emanate from a node.

node n_{ij}: the jth node at depth i of the tree ($1 \leq j \leq Q^i$).

subtree T_{ij}: the subtree whose root node is n_{ij}.

In this chapter, we assume the following communication system.

1) We think of each user as a leaf of a Q-ary tree, as shown in Fig.2.2. The number of users (denoted by N) is assumed to be represented in such a form as Q^K/V, where Q and K correspond to the degree and the depth of a tree graph, respectively. If V=1, every leaf corresponds to a user in the system; otherwise, one of every V leaves corresponds to a user and other leaves never become active.

2) Each user has an address with the form of

$$a_K a_{K-1} \ldots a_1 \quad (0 \leq a_i \leq Q-1, \ i=1,2,\ldots,K). \tag{2.1}$$

For example, in case of Q=4 and K=3, addresses of users are 000, 001, 002, ..., 332 and 333.

3) In each user, one or less packets wait for transmission.

4) The channel time is slotted. A (large) slot consists of a (data) sub-slot and Q mini-slots (see Fig.2.1). A sub-slot is of length equal to a packet transmission time. Whenever a user transmits his packet in a sub-slot, he also sends a signal in one of Q mini-slots according his address. It is assumed that each user can recognize whether a mini-slot contains at least one signal or not; i.e., a mini-slot is busy or empty.

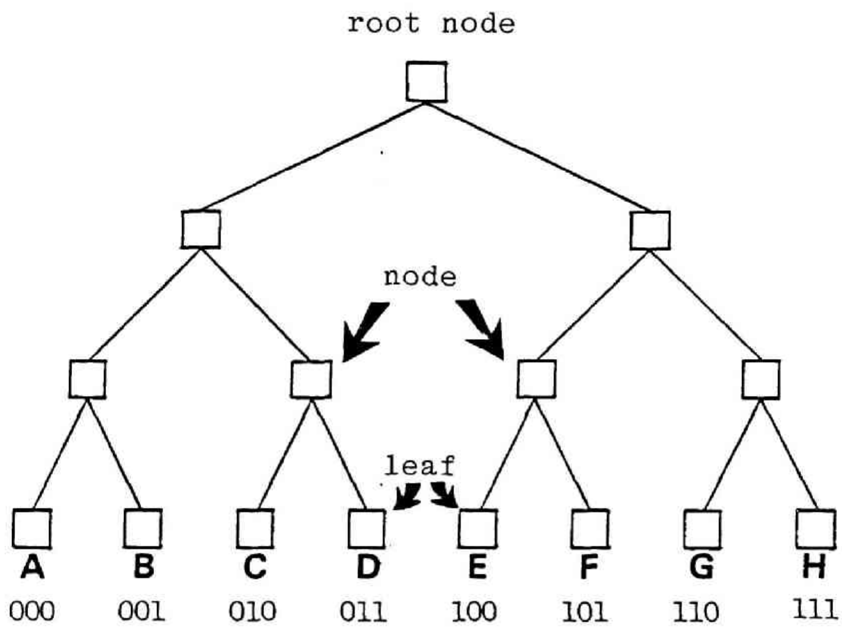


Figure 2.2 A tree graph and terminals in a system

2.3. Adaptive Tree Protocol

We will apply the adaptive tree protocol to a satellite communication system where a propagation delay is too large to be negligible.

2.3.1 Recognition of channel state

As the basic tree protocol, the adaptive tree protocol differentiates between new packets and collided packets; if a slot is used for retransmission of collided packets, newly arriving packets are denied to be transmitted in this slot. Thus, a state of channel slot is classified into two states: contention state, which permits new packets to be transmitted, and reservation state, which is restricted to being used for retransmissions. Each user can recognize a channel state, i.e., whether a channel is in contention or reservation state, by means of the reservation period parameter (RP) which has the following properties:

if RP=0 in the i^{th} slot, the $(i+1)^{st}$ slot becomes contention state,

if RP$\neq$0 in the i^{th} slot, the $(i+1)^{st}$ slot becomes reserved state.

Each user monitors the downlink and detects a collision or success of transmission. If a collision occurs, mini-slots followed by the collided sub-slot can inform each user which group is active, i.e., which group contains at least one packet. Then, the value RP is updated as follows:

1) at the end of slot, RP:=RP+g if a collision is detected,

2) at the start of each slot, RP:=Max(RP-1,0),

where g represents the number of active mini-slots in a collided slot. It is assumed that the value RP of each terminal is identical at any time.

2.3.2 Control procedure

An active user transmits his packet according to the following procedure. Figure 2.3 illustrates an example of the procedure of packet transmission.

Procedure TRANSMISSION

Step 1[initial transmission]. If RP=0 in a slot, a new packet is transmitted in the next slot.

Step 2[transmission and signaling]. In transmitting a packet, the user simultaneously sends a signal to one of Q mini-slots according to his address, i.e., the value of a_{K-i} (see (2.1) in section 2.2); i ($0 \leq i \leq K-1$) of a_{K-i} represents the number of retransmissions experienced by the packet. As an example, we consider the user with address 201. He sends a signal in the 2^{nd} mini-slot in the initial transmission. Thus, in Fig.2.3, the 2^{nd} mini-slot is busy and other mini-slots are empty in slot (1). If a collision occurs, the 0^{th} mini-slot is chosen in the first retransmission. If a collision further arises in the first retransmission, the 1^{st} mini-slot is used in the second retransmission. Even if the second retransmission also results in collision, the third retransmission will surely succeed. In this system, a collided user requires at most three retransmissions (i.e., three partitionings) because each user has a unique address of three figures.

Step 3[observation of the downlink]. After transmitting a packet, a user monitors the downlink. If his packet is correctly received after a round-trip propagation delay, this procedure terminates; otherwise, go to step 4.

Step 4[waiting time for retransmission]. If a user detects a collision of his packet, he will defer his retransmission for w (slots) given by

$$w=m+n,$$

where

m: the number of reserved slots for the backlogged packets, which is given by the value of RP just before the collision is detected,

n: the order of retransmission of his packet among packets collided in the same slot.

For example, in Fig.2.3, a collision arises in slot (1) and is detected in slot (7). In this case, $m=0$ and $n=1$ for user with address 300. Thus he waits for one slot and retransmits his packet in slot (9).

Go back to step 2. □

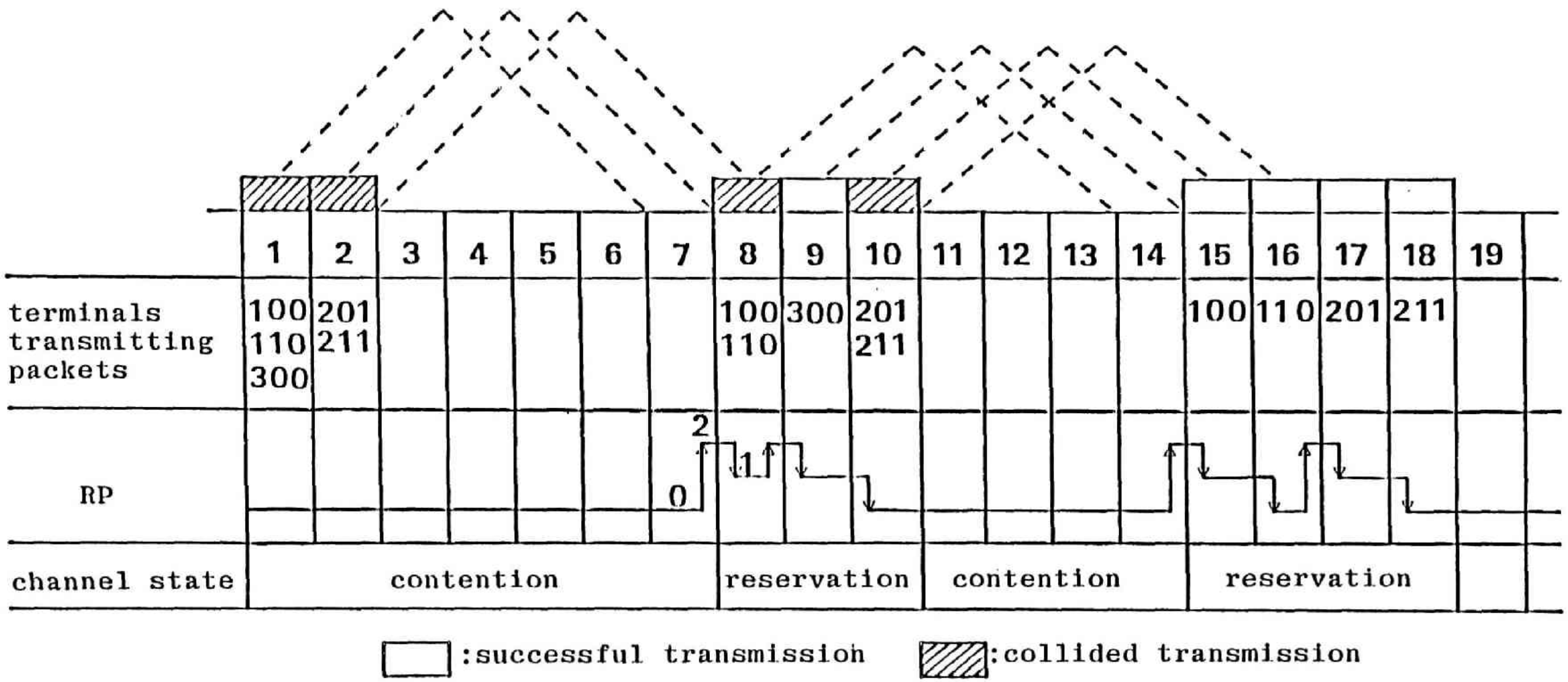


Figure 2.3 Example of transmission process in the adaptive tree protocol (Q=4, K=3)

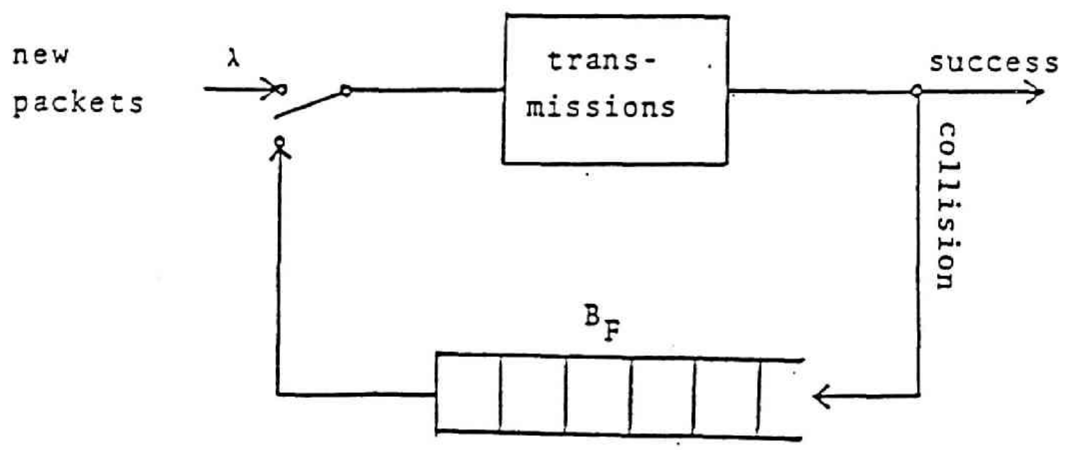
2.4. Average Transmission Delay Analysis

In this section, we analyze the average transmission delay. We can regard the system as the queueing model illustrated in Fig.2.4, in which the virtual common buffer B_F for previously collided packets is assumed to have infinite capacity. This virtual common buffer is introduced as the substitution for all buffers each of which is provided for each user for his collided packet.

2.4.1. Average reservation period and contention period

A channel state is either in contention or reservation state, and thus it becomes alternate sequence of a contention period (C_p^i, i=1,2,...) and a reservation period (R_p^i, i=1,2,...). A time interval consisting of C_p^i and R_p^i is called a cycle. Here it is assumed that C_p^is (i=1,2,...) are independently and identically distibuted, and R_p^is (i=1,2,...) are also independently and identically distributed. Let $\bar{R}_p$ and $\bar{C}_p$ denote the average reservation period and average contention period, respectively.

Whenever a collision arises, the users involved in the collision are partitioned into Q subgroups. In the basic tree protocol (TP), a slot is assigned to each subgroup, so that Q slots are reserved for retransmissions at this time. On the other hand, in the adaptive tree protocol (ATP), a slot is assigned to only an active subgroup. Here, we denote by $\bar{g}$ the average number of such reserved slots; in the basic tree protocol, $\bar{g}$=Q, and in the adaptive tree protocol, $\bar{g}$ is equal to the average number of busy mini-slots in a collided slot. Further, let q_c denote the probability that a collision arises in a slot. The reservation period and the contention period stated above can be regarded as an busy period and an idle period, respectively, in a single-server queueing system in which the probability of a customer arriving during a unit time (slot) is q_c and a mean service time is $\bar{g}$ (slots).



B_F: virtual common buffer for collided packets

Figure 2.4 Queueing model for the adaptive tree protocol

Thus, $\bar{C}_p$ and $\bar{R}_p$ are given by [KLEI 75] as follows:

$$\bar{C}_p = \sum_{i\geq 1} i(1-q_c)^{i-1} q_c = 1/q_c, \tag{2.2}$$

$$\bar{R}_p = \bar{g}/(1-q_c\bar{g}). \tag{2.3}$$

2.4.2 Probability q_c

Under the condition that users transmit their packets with probability p, we denote by q(p) the expected total number of collisions which will arise until all the packets are transmitted successfully. Then, we have (see Appendix 2-A)

$$q(p) = \sum_{i=0}^{K-1} \{Q^i - Q^i(1-p)^{x(K-i)} - Np(1-p)^{x(K-i)-1}\}, \tag{2.4}$$

where x(i) (note that x(K)=N) is defined by

$$x(i) = \begin{cases} Q^i/V & (1\leq i\leq K) \\ 1 & (i=0). \end{cases} \tag{2.5}$$

Considering new packet transmission rate in a slot, we shall devide slots in contention state into two classes: the first slot in contention state, which is the slot marked with B in Fig.2.5, and the other slots, which are the slots marked with A in Fig.2.5. Here, let us introduce the following system parameters:

α : a transmission rate in slot A,

β : a transmission rate in slot B,

λ : a total input rate per sub-slot in the system,

h : a ratio of the mini-slot length to the sub-slot length.

Obviously, the length of a slot is given by 1+Qh in terms of the sub-slot length. Thus, and are given by the following equations:

$$\alpha = \lambda(1+Qh)/N, \tag{2.6}$$

$$\beta = \alpha(1+\bar{R}_p). \tag{2.7}$$

Since that each cycle is independent of each other due to the assumptions

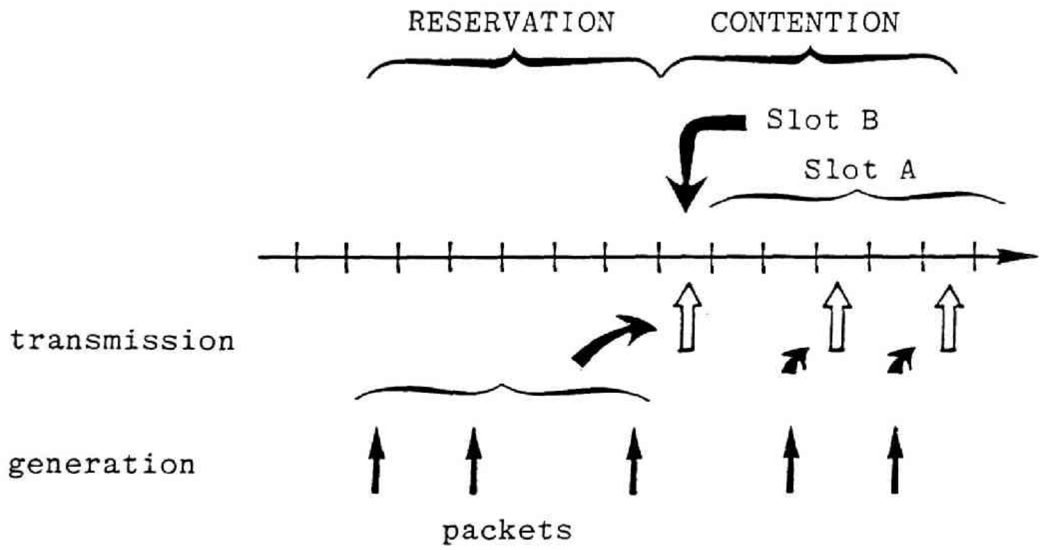


Figure 2.5 Initial transmissions in a contention slot

stated previously, the sum of $q(p)$ over all the slots of a contention period, in steady state, is equal to the average number of collisions occurred in a cycle. Thus the probability q_c, i.e., the probability that a slot suffers a collision, is given by

$$q_c=\{q(\alpha)(\bar{C}_p-1)+q(\beta)\}/(\bar{R}_p+\bar{C}_p).$$

2.4.3 Average number of assigned slots $\bar{g}$

In this section, we derive the average number, $\bar{g}$, of reserved slots in a collision; i.e., the average number of busy mini-slots in a collided slot. In a tree graph, we refer to a tree with at least two active leaves as a colliding tree, and a tree with at least one active leaf as an active tree. Then, we will obtain the probability that a colliding tree rooted at node n_{ij} contains i active subtrees. We denote this probability by $g_i^j(p)$ for a given value of p. Then, $g_i^j(p)$ is given by

$$g_i^j(p)=\text{Prob}[\text{i subtree } T_{j+1}\text{s are active} \mid T_j \text{ is colliding}],$$

where T_j represents a subtree whose root node is at depth j. We further introduce the following probabilities:

$$P_a^j(p)=\text{Prob}[T_j \text{ is active}]$$
$$=1-(1-p)^{x(K-j)},$$

$$P_c^j(p)=\text{Prob}[T_j \text{ is colliding}]$$
$$=1-(1-p)^{Q(K-j)}-x(K-j)p(1-p)^{x(K-j)-1},$$

Thus, using $P_a^j(p)$ and $P_c^j(p)$, $g_i^j(p)$ is given by

$$g_i^j(p)=\begin{cases}\binom{Q^*}{i}X^i(1-X)^{Q^*-i}/P_c^j(p) & (i\geq 2)\\ \{Q^*P_c^{j+1}(1-X)^{Q^*-1}\}/P_c^j(p) & (i=1),\end{cases}$$

where X stands for $P_a^{j+1}(p)$ and Q^* satisfies the following equation:

$$Q^*=\begin{cases}Q & (0\leq j\leq K-2)\\ Q/V & (j=K-1).\end{cases}$$

Note that $g_i^{K-1}(p)=0$ $(Q/V<i\leq Q)$ if $V\neq 1$.

Next, we define the following probability:

$g_i(p)$=Prob[i mini-slots are busy in a collided slot].

In order to obtain this probability, we will try to remove the condition on depth j in $g_i^j(p)$. Since that the probability that a slot corresponding to a node at depth j is collided is given by $Q^jP_c^j(p)/\sum_{j=0}^{K-1} Q^jP_c^j(p)$, and that $\sum_{j=0}^{K-1} Q^jP_c^j(p)$ is equal to $q(p)$, $g_i(p)$ is given by

$$g_i(p)=\{\sum_{j=0}^{K-1} Q^jP_c^j(p)g_i^j(p)\}/q(p).$$

By defining the following probability generating function:

$$G_p(Z)=\sum_{i=1}^{Q} g_i(p)Z^i,$$

$\bar{g}(p)$ is given by differentiating $G_p(Z)$ with respect to Z; i.e.,

$$\begin{aligned}\bar{g}(p)&=G_p'(1)\\&=f(p)/q(p),\end{aligned}$$

where

$$f(p)=q(p)+Np-1+(1-p)^N.$$

In a way similar to the derivation of q_c in section 2.4.2, we can obtain

$$\bar{g}=\{f(\alpha)(\bar{C}_p-1)+f(\beta)\}/\{q(\alpha)(\bar{C}_p-1)+q(\beta)\}. \tag{2.8}$$

Next, let $\overline{g^2}$ denote the second moment of the number of assigned slots. Then, $\overline{g^2}$ is given by

$$\overline{g^2}=\{\overline{g^2}(\alpha)q(\alpha)(\bar{C}_p-1)+\overline{g^2}(\beta)q(\beta)\}/\{q(\alpha)(\bar{C}_p-1)+q(\beta)\}$$

where $\overline{g^2}(p)$ is the second moment of $g_i(p)$ and is given by the following equation:

$$\overline{g^2}(p)=G_p''(1)+G_p'(1).$$

2.4.4 Average number of retransmissions $\bar{r}$

We will derive r(p) which is the average number of retransmissions for a given value of p (where p represents transmission rate per slot). Obviously, for a given p, the average number of retransmissions experienced by a packet in the adaptive tree algorithm is equal to that of the basic tree algorithm. Let $r_i(p)$ denote the probability that a packet is retransmitted i times prior to successful transmission for a given p, and $r(p)$ denote the average number of retransmissions; $r(p)$ is given in terms of $r_i(p)$ as

$$r(p)=\sum_{i=0}^{K} i r_i(p). \tag{2.9}$$

Suppose that an active user, say user t_a, which belongs to subtrees $T_{(i-1)x}$ and T_{iy} (see section 2.2 for the definitions), has already retransmitted a packet (i-1) times and the next retransmission is successful. Then $r_i(p)$ is given by

$r_i(p)$=Prob[T_{iy} contains no active users except t_a]

·Prob[at least one active user is involved in the (Q-1) subtree T_{iz}s except T_{iy} which belong to $T_{(i-1)x}$],

so that $r_i(p)$ is explicitly given by the following:

$$r_i(p)=\begin{cases}(1-p)^{x(K-i)-1}(1-(1-p)^{x(K-i)(Q-1)}) & (i\geq 1)\\ (1-p)^{x(K)-1}. & (i=0)\end{cases} \tag{2.10}$$

The average number of retransmissions $r(p)$ is given by (2.9) and (2.10).

Further, we introduce the following probabilities:

P_A=Prob[a user initially transmits his packet in slot A]

$=(\bar{C}_p-1)/(\bar{R}_p+\bar{C}_p)$.

P_B=Prob[a user initially transmits his packet in slot B]

$=(\bar{R}_p+1)/(\bar{R}_p+\bar{C}_p)$.

Since a transmission rate per slot is α in slot A and β in slot B, we can obtain $\bar{r}$ as follows:

$\bar{r}=r(\alpha)P_A+r(\beta)P_B$

2.4.5 Average waiting time for retransmission $\bar{W}$

In this section, we discuss the waiting time spent by a packet in a virtual common buffer B_F (see Fig.2.4). Since this waiting time corresponds to w at step 4 of Procedure TRANSMISSION in section 3.2, the average waiting time is also consists of two parts; one is incurred by service time for packets in a common buffer when a packet arrives (denote this waiting time by $\bar{W}_1$), and the other is incurred by waiting time of its turn for retransmission among the collided packets in the same slot (denote this waiting time by $\bar{W}_2$). In order to obtain $\bar{W}_1$, we will apply the average waiting time formula for a G/G/1 queueing system [KLEI 75b] to the discrete time queueing model described in section 2.4.1. Then, $\bar{W}_1$ is given as follows:

$$\bar{W}_1=\{\sigma_a^2+\sigma_b^2+(\bar{t})^2(1-\rho)^2\}/\{2\bar{t}(1-\rho)\}-\overline{I^2}/(2\bar{I}),$$

where

σ_a^2: variance of interarrival time

$=(1-q_c)/q_c^2,$

σ_b^2: variance of service time

$=0$ (TP), $\overline{g^2}-(\bar{g})^2$ (ATP),

$\bar{t}$: mean interarrival time

$=(1-q_c)/q_c,$

$\bar{I}$: mean idle period

$=1/q_c,$

$\overline{I^2}$: second moment of idle period

$=(2-q_c)/q_c^2,$

ρ : utilization factor

$=q_cQ$ (TP), $q_c\bar{g}$ (ATP).

As for $\bar{W}_2$, we immediately obtain the following result:

$$\bar{W}_2=(\bar{g}-1)/2.$$

2.4.6 Average packet delay time $\bar{D}$

A delay time experienced by a packet consists of the following three delay times. The first (denoted D_1) is the time between its generation and the initial transmission; and the second (denoted D_2) is initial transmission time and a round trip propagation delay; and the last (denoted D_3) is the time for retransmissions. We denote the average values of D_1 and D_3 by $\bar{D}_1$ and $\bar{D}_3$, respectively. Then, $\bar{D}_1$ and $\bar{D}_3$ are given by

$$\bar{D}_1=(1/2)P_A+\{(1+\bar{R}_p)/2\}P_B,$$

$$\bar{D}_3=\bar{r}(1+R+\bar{W}).$$

In the above equations, R represents a round trip propagation delay (slots) given by

$$R=\lceil R_0/(L/S)\rceil \text{ (slots)},$$

where $\lceil x\rceil$ denotes the minimum integer greater or equal to x and

R_0=(a round trip propagation delay) (sec),

S =(channel capacity) (bit/sec),

L =(slot length) (bit).

Thus, the average packet delay time $\bar{D}$ is given by

$$\bar{D}=\bar{D}_1+D_2+\bar{D}_3,$$

where $D_2=1+R$.

Finally, it is necessary to determine $\bar{C}_p$, $\bar{R}_p$, q_c and $\bar{g}$ in order to obtain $\bar{D}$. We will execute the following procedure on the assumption that λ, h, Q, N, K and R are given.

Procedure CALCULATE

Step 1. Substituting Eqs. (2.2) and (2.3) into Eq. (2.7), we get a quadratic equation with respect to q_c. The positive solution is given by

$$q_c=\{-X+\sqrt{X^2+4\bar{R}_p q(\alpha)}\}/(2\bar{R}_p). \tag{2.11}$$

where $X=1+q(\alpha)-q(\beta)$. Note that the other solution is negative and meaningless in our analysis.

Step 2. (2.3) can be rewritten as

$$\bar{g}=\bar{R}_p/(1+q_c\bar{R}_p). \qquad (2.12)$$

Substituting Eqs. (2.2) and (2.12) into Eq. (2.8), we get another quadratic equation with respect to q_c. The positive solution is given by

$$q_c=\{-Y+\sqrt{Z}\}/[2\{f(\beta)-f(\alpha)\}\bar{R}_p], \qquad (2.13)$$

where

$$Y=f(\beta)-f(\alpha)(1-\bar{R}_p)+\bar{R}_p\{q(\alpha)-q(\beta)\},$$

$$Z=Y^2-4\bar{R}_p\{f(\beta)-f(\alpha)\}\{f(\alpha)-q(\beta)\bar{R}_p\}.$$

Step 3. Obtain $\bar{R}_p$ and q_c by solving Eqs. (2.11) and (2.13). Furthermore, by substituting these $\bar{R}_p$ and q_c into Eq. (2.12), we get $\bar{g}$.

Halt. □

The average transmission delay can be expressed in terms of sub-slot time length as follows:

$$\bar{D}_S=\bar{D}(1+Qh).$$

2.5. Numerical and Simulation Results

We assume a satellite communication system such as 50 (Kbit/sec) satellite channel with 270 (ms) round-trip propagation delay and 1125 bit data packet according to [SCHW 77], so that R is equal to 12 frames in case of h=0. In order to compare the ATP with the TP, we use a sub-slot length, which is equal to a packet transmission time, instead of (large) slot as a unit of channel time. We denote a total channel input rate per sub-slot by λ.

We give numerical results for various values of Q and λ in case that the population of terminals is 1024 and each terminal has one or less packets at any time in Figs. 2.6 and 2.7. Figure 2.6 illustrates the average transmission delay for the TP, and Fig. 2.7 shows that for the ATP. Figures 2.6 and 2.7 show that the approximate analysis is well agree with simulations. These figures further indicate that the improvement is achieved by introduction of mini-slots. Figure 2.7 shows that the ATP achieves much higher maximum throughput and lower packet delay time than the TP.

Table 2.1 shows numerical results due to our approximate analysis and simulation results. In the simulations, it is assumed that each user is capable of containing five packets and new packets independently arrive at each user according to a Poisson process. Note that all the arriving packets entered each terminal without rejection in our simulations. In Table. 2.1, two kinds of delay times are given: an average service time which is equal to $D_2+\bar{D}_3$ in section 2.4.6, and an average system time which represents $\bar{D}$ ($=\bar{D}_1+D_2+\bar{D}_3$) in the same section. This table indicates that numerical results for an average service time are well agree with those simulation results even at high input rate. For an average system time (i.e., an average packet delay time), numerical results are well agree with those simulation results except at heavy loads. To make the analysis more precise, we must analyze a behavior of packets waiting for initial transmission at heavy loads. Furthermore, it can be seen from Table 2.1 that the delay time performance is not so sensitive to the number of users in a system.

Next, we investigate the effect of the overhead required per frame. Since a busy and an empty state are all information on the state in a mini-slot, it can be considered that a mini-slot requires a few or at most several bits. The delay vs. channel input performances for h=0.005 and h=0.01 are illustrated in Figs. 2.8 and 2.9, respectively. As Q becomes large and larger, the overhead has a deteriorating effect on the delay vs. channel input performance. Comparing Fig. 2.9 with Fig. 2.8, we find little effect for Q=2 and 4 but the maximum throughput decreases by about five percent for Q=8 and about ten percent for Q=16. Nevertheless, the ATP performs better than the TP. For a specific value of h, we can find an optimum value of degree that provides the best delay vs. channel input performance.

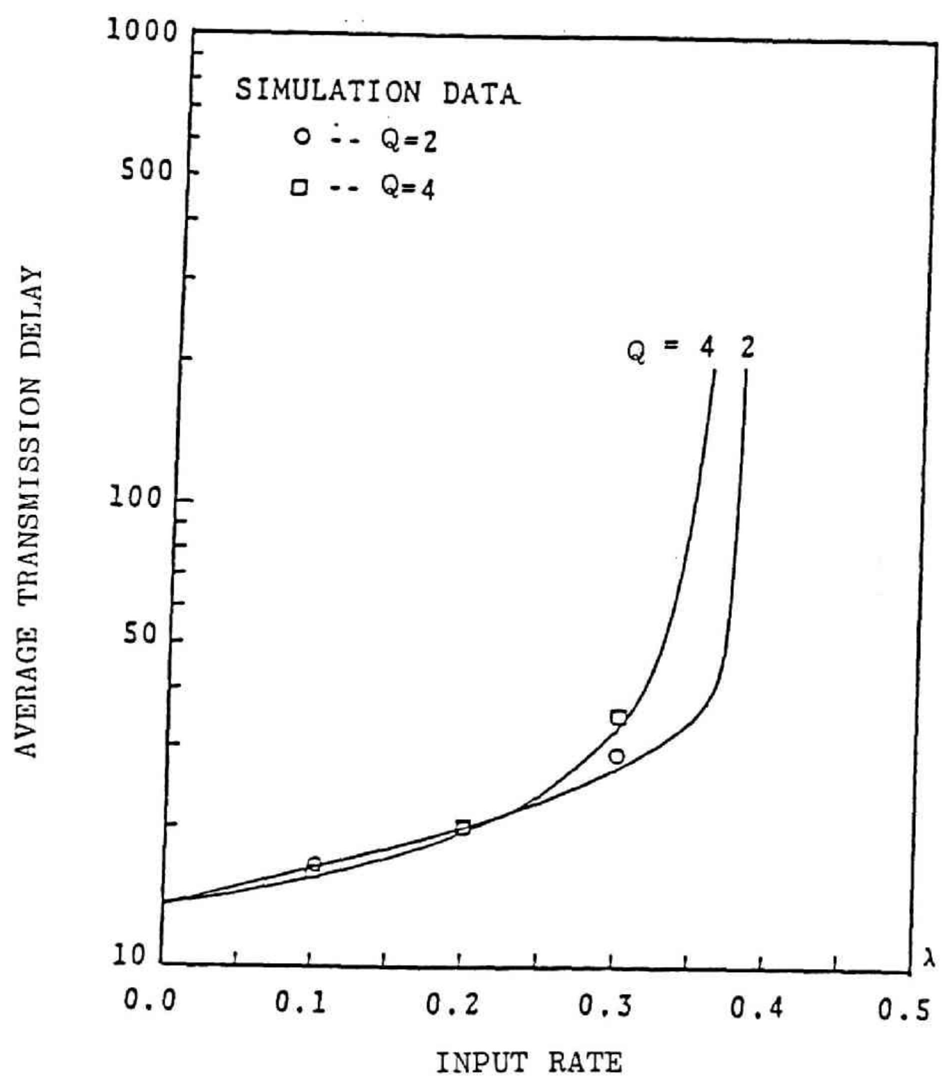


Figure 2.6 Average transmission delay of the tree protocol (N=1024)

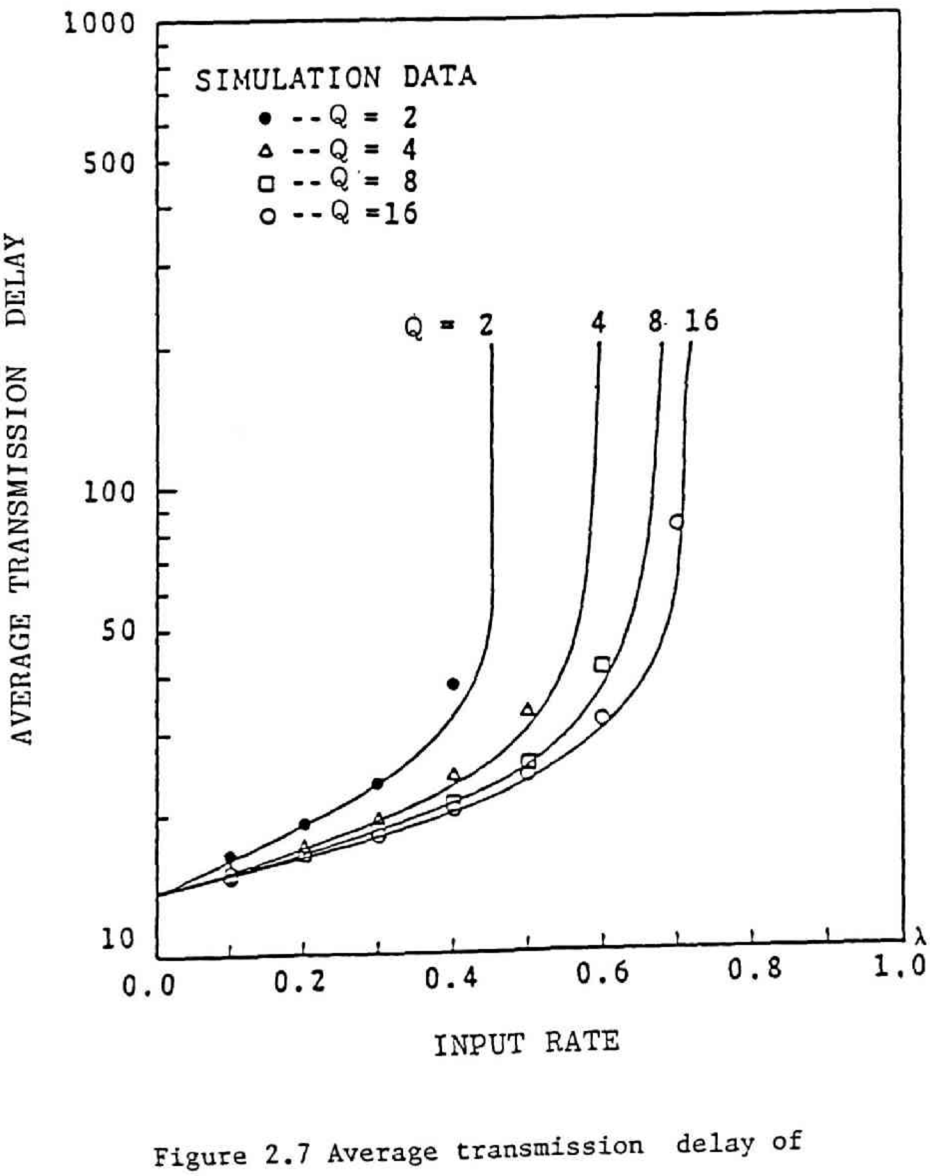


Figure 2.7 Average transmission delay of the adaptive tree protocol (N=1024, h=0.0)

Table 2.1

Numerical and simulation results
for transmission delay
in the adaptive tree protocol (Q=8, h=0.0)

Numerical results (Simulation results)

λ \ N		4096	1024	512
0.1	A	14.51	14.51(14.55)	14.50(14.23)
	B	15.03	15.02(15.05)	15.02(14.77)
0.2	A	16.12	16.11(15.91)	16.10(15.90)
	B	16.68	16.67(16.50)	16.66(16.55)
0.3	A	18.02	18.00(17.90)	17.98(17.82)
	B	18.68	18.67(18.62)	18.65(18.61)
0.4	A	20.51	20.48(20.13)	20.45(20.05)
	B	21.37	21.34(21.10)	21.34(21.13)
0.5	A	24.32	24.25(23.91)	24.19(23.80)
	B	25.59	25.53(25.66)	25.45(25.73)
0.6	A	32.73	32.48(33.71)	32.20(33.61)
	B	35.45	35.18(39.01)	34.84(39.54)
0.65	A	59.47	54.43(58.93)	50.15(51.69)
	B	67.93	61.18(83.72)	56.75(72.28)

N : the number of terminals,
A : $D_2+\bar{D}_3$, B : $\bar{D}_1+D_2+\bar{D}_3$.

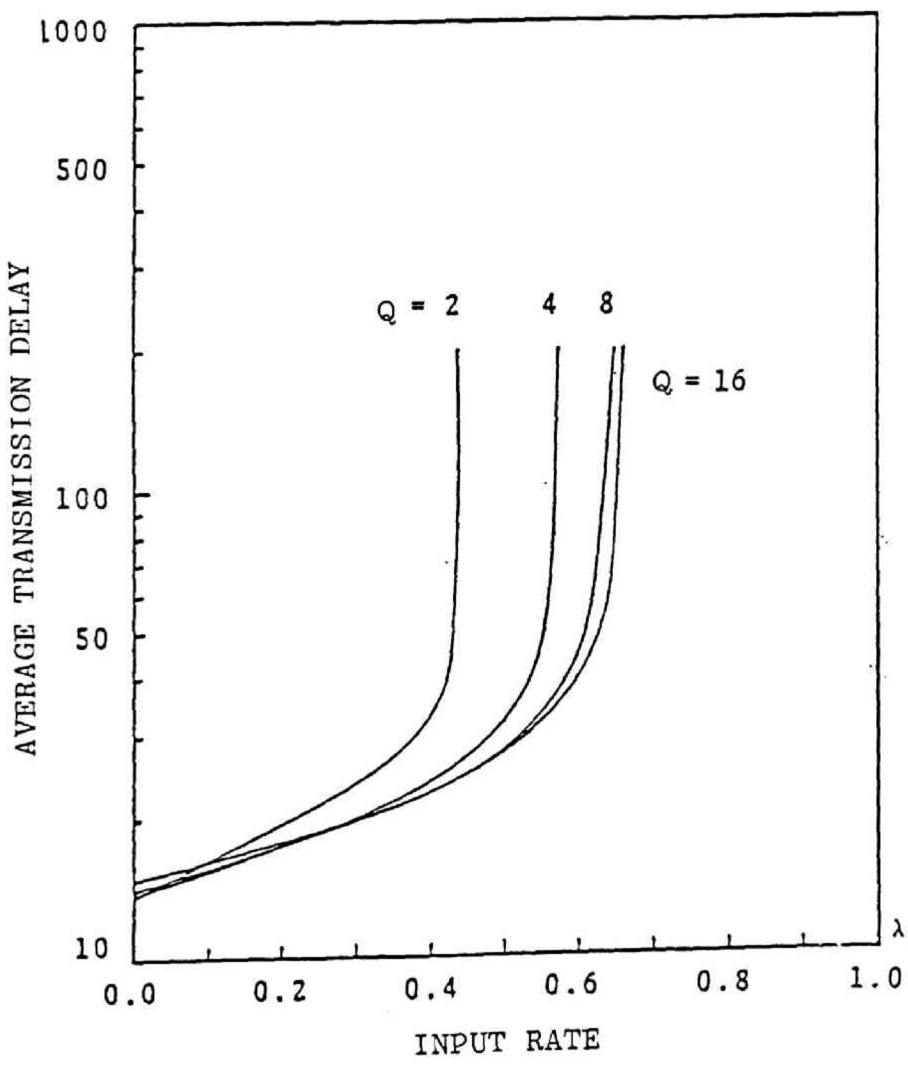


Figure 2.8 Average transmission delay of the adaptive tree protocol (N=1024, h=0.005)

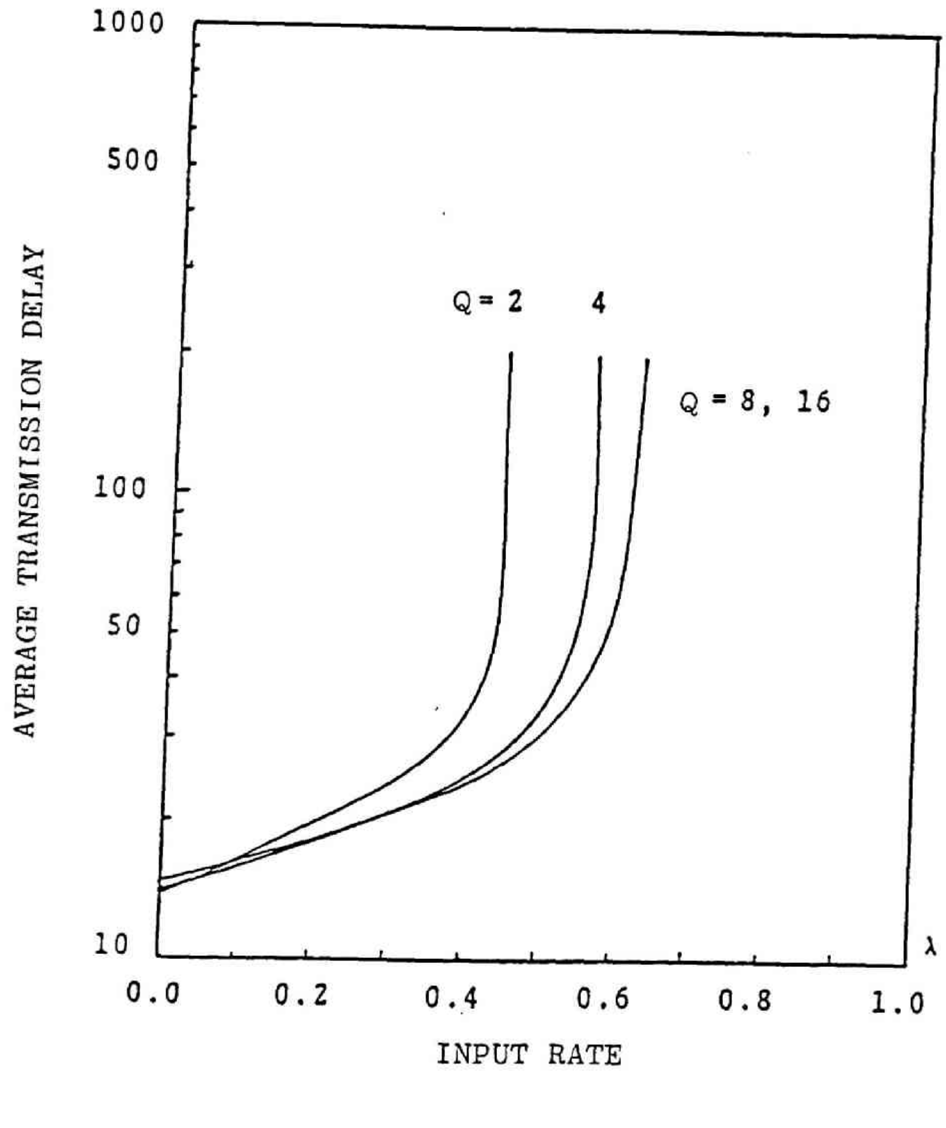


Figure 2.9 Average transmission delay of

the adaptive tree protocol (N=1024, h=0.01)

2.6. Conclusion

In this chapter, we proposed the adaptive tree protocol in order to improve the performance of the basic tree protocol. In the Q-ary basic tree algorithm, if a collision arises, the packets involved in the collision are partitioned into Q subgroups and one slot is assigned to each subgroup. On the other hand, the adaptive tree protocol assigns slots for retransmissions not to every group but only to each active group in case of channel collision. It has turned out from our analysis that the adaptive tree protocol performs fairly efficient compared to the basic tree protocol. The performance of the adaptive tree protocol becomes better as the degree Q increases. The sensitivity of the degree to the performance was discussed in section 2.5. Consequently, we can conclude that, owing to its low average delay time and its high channel capacity, the adaptive tree protocol is an efficient protocol in random access communication systems.

APPENDIX 2-A: Derivation of Eq.(2.4)

We consider the behaviour of all the packets whose initial transmissions occur in the same slot. For fixed transmission rate per slot, p, let $q^j(p)$ denote the random variable representing the sum of collided slots (from at depth 0 to at depth j-1) due to packets transmitted at depth j in a tree graph. For the case of collision, we define the following:

$$q(i,j)=\mathrm{Prob}[q^j(p)=i-1] \quad (i\geq 2).$$

Furthermore, for the case of non-collision, we define

$$q(0,j)=(1-p)^{x(j)},$$

$$q(1,j)=x(j)p(1-p)^{x(j)-1},$$

where x(i) is defined by Eq.(2.5). The maximum number of collided slots is equal to whole sum of nodes except leaves in a tree graph, then the following equation is obtained,

$$\sum_{i=0}^{y(j)+1} q(i,j)=1, \text{ (where } y(j)=\sum_{n=0}^{j-1} Q^n).$$

Now, we define the probability generating function of q(i,j) as follows:

$$U_j(z)=\sum_{i=0}^{y(j)+1} q(i,j)z^i.$$

Then, we can obtain the following recursive equation for $U_j(z)$ in such a way used in [HAYE 78] under the assumption that a packet arrival in each of Q^j/V terminals is independent and identically distributed:

$$\begin{aligned} U_j(z)=&q^Q(0,j-1)+Qq(1,j-1)q^{Q-1}(0,j-1)z \\ &+[\sum_{n=0}^{Q}\binom{Q}{n}q^n(0,j-1)\{\prod_{k=1}^{Q-n}\sum_{i_k=1}^{y(j-1)+1} q(i_k,j-1)z^{i_k-1}\}z^2 \\ &-q^Q(0,j-1)z^2-Qq^{Q-1}(0,j-1)q(1,j-1)z^2]. \end{aligned} \qquad (2A.1)$$

The first term in the right hand side of Eq. (2A.1) is the probability that Q subtrees with Q^{j-1}/V terminals do not involve any packet. The second term implies the successful transmission. The term in brackets [ ] corresponds to the case that at least two subtree are active at depth j-1; i.e., the case of collided transmission. After some manipulation, Eq. (2A.1) becomes

$$U_j(z)=q^Q(0,j-1)(1-z^2)+Qq(1,j-1)q^{Q-1}(0,j-1)(1-z)z$$
$$+[\sum_{n=0}^{Q}\binom{Q}{n}q^n(0,j-1)z^n\{U_{j-1}(z)-q(0,j-1)\}^{Q-n}]z^{2-Q} \qquad (j\geq 2),$$

hence

$$U_j(z)=q^Q(0,j-1)(1-z^2)+Qq(1,j-1)q^{Q-1}(0,j-1)(1-z)z$$
$$+[U_{j-1}(z)-q(0,j-1)(1-z)]^Q \qquad (j\geq 2). \tag{2A.2}$$

As for $U_1(z)$, we consider a tree graph which involves only Q/V leaves, and it is given by

$$U_1(z)=[1-(1-p)^{x(1)}-x(1)p(1-p)^{x(1)-1}]z^2+(1-p)^{x(1)}+x(1)p(1-p)^{x(1)-1}z. \tag{2A.3}$$

The relation between q(p) and $U_K(z)$ is given by the following equation:

$$q(p)=\sum_{i=1}^{y(K)} iq(i+1,K)$$
$$=\frac{d}{dz}U_K(z)|_{z=1}-(1-q(0,K))$$
$$=U_K'(1)-[1-(1-p)^N]. \tag{2A.4}$$

We can obtain the following recursive relationships for $U_j'(1)$ by differentiating Eqs. (2A.2) and (2A.3):

$$U_j'(1)=Q[U_{j-1}'(1)+(1-p)^{x(j-1)}]-2(1-p)^{x(j)}-x(j)p(1-p)^{x(j)-1}+2-Q \quad (j\geq 2),$$

$$U_1'(1)=2-2(1-p)^{x(1)}-x(1)p(1-p)^{x(1)-1}.$$

Furthermore, the explicit expression for q(p) is obtained from Eq. (2A.4) by substituting successively; i.e.,

$$q(p)=\sum_{i=0}^{K-1}[Q^i-Q^i(1-p)^{x(K-i)}-Np(1-p)^{x(K-i)-1}].$$

Chapter 3

Throughput Analysis of Blocked Access Tree algorithms with Mini-Slots

3.1. Introduction

This chapter treats a class of protocols in a multiple access environment. The system to be considered is characterized by such properties that channel time is slotted, simultaneous transmission of more than one packet leads to a collision, and none of collided packets are correctly received. We further assume noiseless channel.

In the context of multiple access, a collision resolution algorithm (CRA) plays a key role in scheduling a large number of users to transmit their packets on a single common channel. In particular, the tree-based CRAs have extensively studied because of their stable behavior (see, e.g., section 1.4.1). Recently, some of these studies have exploited additional feedback information in order to enhance performance of tree-based CRAs. So far two kinds of additional feedback information have introduced; one is collision multiplicity, which is available through energy detectors, and the other is the information provided by (control) mini-slots. Through the former information, Tsybakov [TSYB 80b] and Georgiadis et al. [GEOR 83] obtained maximum throughput 0.5324. The latter was exploited in several papers such as [BERG 83, MERA 83, HUAN 83, HUAN 85].

In a class of tree algorithms with mini-slots (TA/Ms), Q mini-slots are provided within a (large) slot to allow users to acquire additional information on the state of the packet transmission (see Fig.3.1). Data sub-slot length is equal to packet transmission time. We assume in this chapter that new packets are denied to be transmitted until all the outstanding packets are transmitted correctly (see, e.g., [MATH 85]);

such a protocol is referred to as the blocked access class (a TA/M in the other class, i.e., a free access class, will be considered in the following chapter). When a user sends a packet (using a data sub-slot in a large slot), he also sends a signal in a mini-slot randomly chosen (in chapter 2, a user was assumed to choose a mini-slot according to his address).

This class of algorithms proposed so far are broadly divided into two subclasses in accordance with available feedback information on the state of mini-slot: binary feedback and ternary feedback. In binary feedback case, users can only distinguish between an empty mini-slot (i.e., no signal detected) and a busy mini-slot (i.e., signal detected). In ternary feedback case, users further can distinguish between a successful mini-slot (i.e., only one user is sending a signal) and a collided mini-slot (i.e., more than one user is sending a signal) in a busy mini-slot. Both the adaptive tree algorithm presented in chapter 2 and the multibit feedback algorithm (MFA) proposed in [BERG 83, HUAN 85] belong to the binary feedback case. The ternary feedback case was considered in [MERA 83, HUAN 83]; Merakos has shown that the binary TA/M with ternary feedback, referred to as the left right tree algorithm (LRTA), attains a maximum throughput of 0.513. The performance of TA/Ms heavily depends on the number of mini-slots Q. Thus, the main issue in this class of algorithms is to analyze the performance in terms of Q, but such analysis has not yet been carried out.

In this chapter, we focus our attention on two blocked access (Q-ary) tree algorithms with mini-slots and obtain the stable maximum throughput, which represents the maximum value (thus, critical value) of throughput that provides the stable channel; in a mini-slot, the first algorithm assumes that binary feedback information is available, and the second algorithm assumes ternary feedback information. The exact description of the algorithms is presented in section 3.2. In section 3.3, we will derive the average conditional collision resolution time (CRT) for a given value of collision multiplicity for above algorithms; this quantity characterizes the performance of these algorithms. In section 3.4, by analyzing the asymptotic behavior of the average conditional CRT when the collision multiplicity approaches infinity, we will obtain the stable maximum throughput of TA/Ms as a function of Q. In section 3.5, we will show an optimum value of Q maximizing throughput of a TA/M-BF for a given mini-slot length. Other numerical results and simulation results are provided.

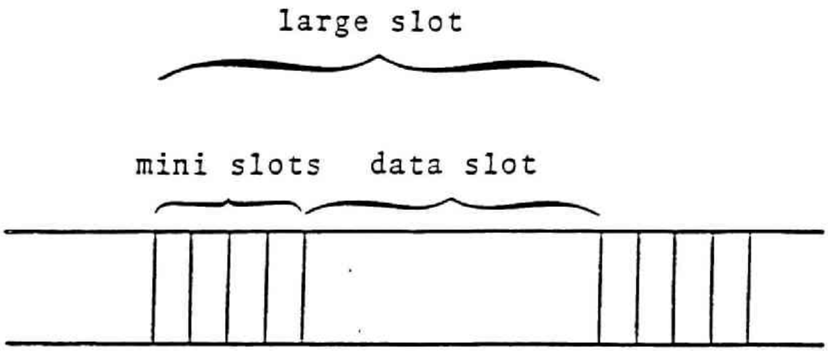


Figure 3.1 Slot configuration

3.2 Blocked Access Q-ary Tree Algorithms with Mini-Slots

In our analysis, we assume that the channel time is slotted, and a (large) slot consists of Q number of mini-slots and a data sub-slot (Fig.3.1). The size of a data sub-slot is equal to transmission time of a packet. The propagation delay is assumed to be zero; this assumption has no effect on the throughput performance.

We consider a Blocked Access Q-ary Tree Algorithm with Mini-slots (BA TA/M). Note that, in blocked access algorithms, newly arriving packets are forced to wait until all the outstanding collided packets have been removed from a system. Thus, these new packets never interfere with previously collided packets. We treat two cases regarding the feedback information available in mini-slots; ternary and binary feedback information.

BA TA/M with Ternary Feedback (BA TA/M-TF);

Users are able to distinguish between an empty mini-slot (i.e, no user is sending a signal), a successful mini-slot (i.e., only one user is sending a signal) and a collided mini-slot (i.e., more than one user is sending a signal).

BA TA/M with Binary Feedback (BA TA/M-BF);

Users have limited capability to detect the signal level of a mini-slot so that users can only distinguish between an empty mini-slot (i.e., no signal detected) and a busy mini-slot (i.e., signal detected).

In a Q-ary TA/M with ternary feedback information, when a user sends a packet (in a data slot of a large slot), he also sends a signal in a mini-slot randomly chosen. In case of collision, the current enabled set of users are first partitioned into Q number of sub-sets, each corresponding to a group of users who have chosen the same mini-slot. Non-active sub-sets (i.e., sub-sets which correspond to empty mini-slots) are deleted from the further collision resolution process. Each active sub-set with only one active user (i.e., a sub-set which corresponds to a successful mini-slot) constitutes a new subtree. Each active sub-set with more than one active user (i.e., a sub-set which corresponds to a collided mini-slot) is further divided into m subtrees, where each active user is randomly assigned to one of the m new subtrees. (This algorithm, hence, will be referred to as a BA TA/M-TF(m) in the following.) Thus, this division process results in $s+m\times c$ number of new subtrees, where s and c are the numbers of successful and collided mini-slots in a large slot, respectively. One from these new subtrees is chosen for collision resolution in the subsequent (large) slot. If further collision occurs, the enabled set is continually divided in the same manner until the collision is resolved. As for new packets, they wait until all previously arising collisions are resolved. The LRTA [MERA 83] is equivalent to the binary BA TA/M-TF(2).

Note that a Q-ary TA/M with binary feedback is equivalent to a Q-ary TA/M with ternary feedback where m is equal to 1 (i.e., BA TA/M-TF(1)). In a Q-ary BA TA/M-BF, since users cannot distinguish between a successful mini-slot and a collided mini-slot, all the active subtrees, regardless of how many active users exist in each of them, are treated in the same way; this is nothing but how a Q-ary BA TA/M-TF(1) acts. In this class, no empty slot is involved in a <u>collision resolution interval</u> (CRI). The MFA is of this class; it was combined with the interval searching algorithm in [HUAN 85].

In addition, the MFA in [BERG 83] can be regarded as a modified version of BA TA/M-BFs. This algorithm takes a different action from the above BA TA/M-BF only in the case that a collided slot contains only one busy mini-slot. In this case, this busy mini-slot is obviously a collided mini-slot. Thus, in the MFA in [BERG 83], the users having sent packets in such a slot are immediately divided into m (m=2 in [BERG 83]) subgroups, and the users in one of these subgroups will send their packets in the subsequent slot. This algorithm will be referred to as a BA TA/M-BF(m); note that the BA TA/M-BF(1) is identical to the BA TA/M-BF above stated.

In the following, we will particularly pay our attention to the BA TA/M-TF(m) and the BA TA/M-BF.

Figures 3.2 and 3.3 illustrate a collision resolution process in a ternary (i.e., Q=3) BA TA/M-TF(2) and a ternary (Q=3) BA TA/M-BF, respectively. Figure 3.4 is related to a ternary (Q=3) basic tree algorithm (without mini-slots), which will serve as a reference for comparison. In these figures, an initial collision was due to 4 users (A, B,C, D). Users A and B have chosen the first mini-slot to send signal, and C and D have chosen the third mini-slot, resulting in no successful mini-slots and two collided mini-slots; i.e., s=0 and c=2.

In the following, a subtree rooted at node A will be referred to as subtree A for convenience. In the TA/M-TF(2) (see Fig.3.2), the users involved in the initial collision are partitioned into 4 (i.e., $s+c\times m=4$) enabled subtrees; subtree 2 with no active users, subtree 3 with users A and B, subtree 6 with user C, and subtree 9 with user D. Since subtree 2 has no active users, the slot (slot 2) assigned to this subtree remains unused. Subtree 3 results in collision again and are thus again divided for the further collision resolution. Users C and D are isolated in subtrees 6 and 7, respectively, resulting in successful transmissions.

On the other hand, the TA/M-BF partitions the users into 2 (i.e., $s+c\times l=2$) subtrees; subtrees rooted at nodes 2 and 6. Since all the users in the same subtree transmit in the same slot, both subtrees 2 and 6 lead to collision again. The same collision resolution process is repeated until all the outstanding collided packets are removed from the system. In the TA/M-BF, a CRI is free from empty slots, differently from a CRI in the TA/M-TF(m) ($m\geq 2$). As is clear from the mechanism of a TA/M-BF, we can obtain Fig. 3.3 by eliminating empty slots in the CRI in Fig. 3.4.

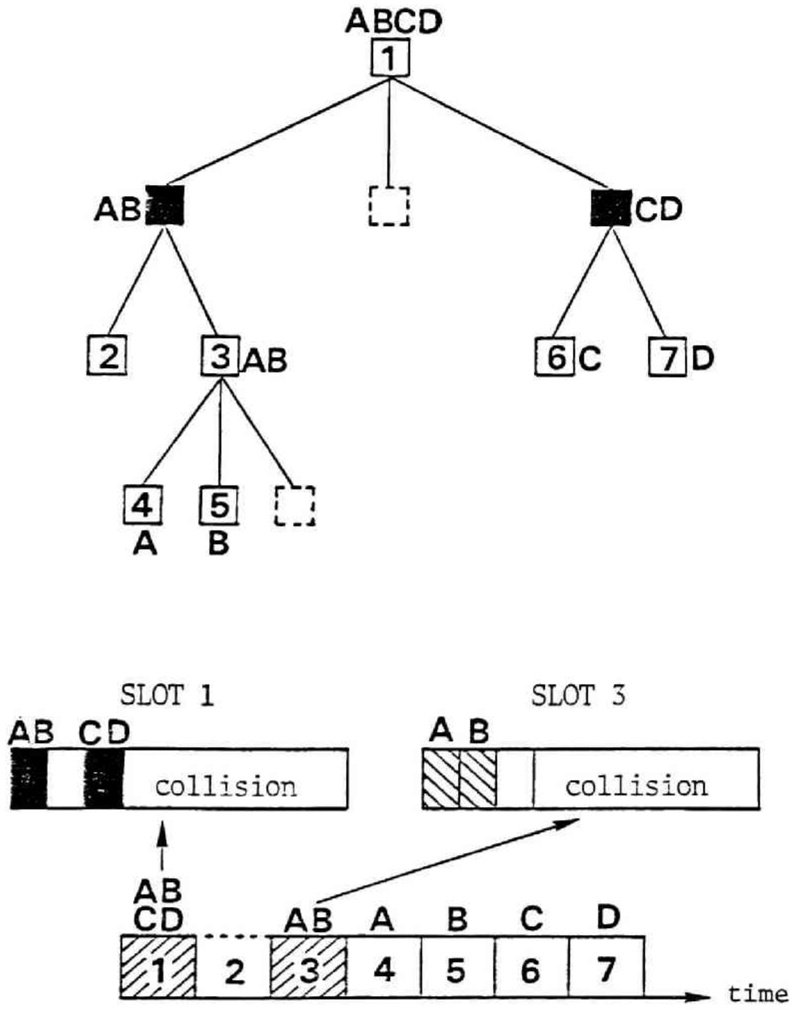


Figure 3.2 Example of collision resolution procedure in TA/M-TF(2) with Q=3

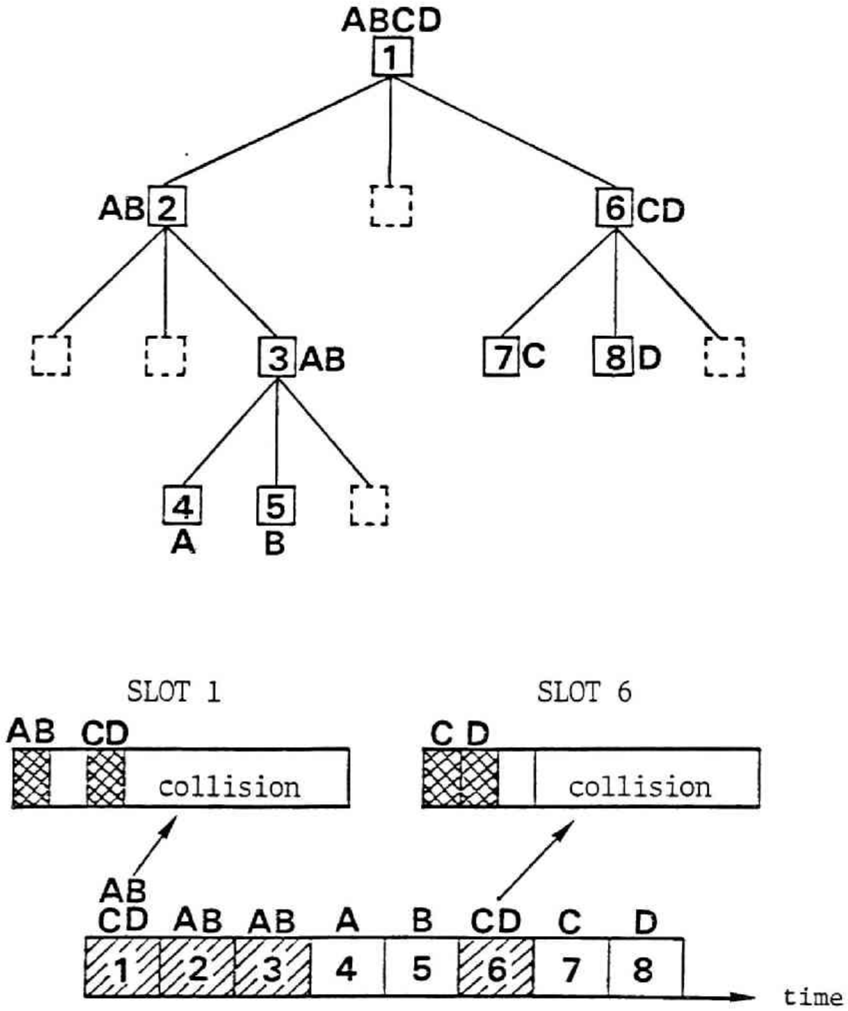


Figure 3.3 Example of collision resolution procedure in TA/M-BF with Q=3

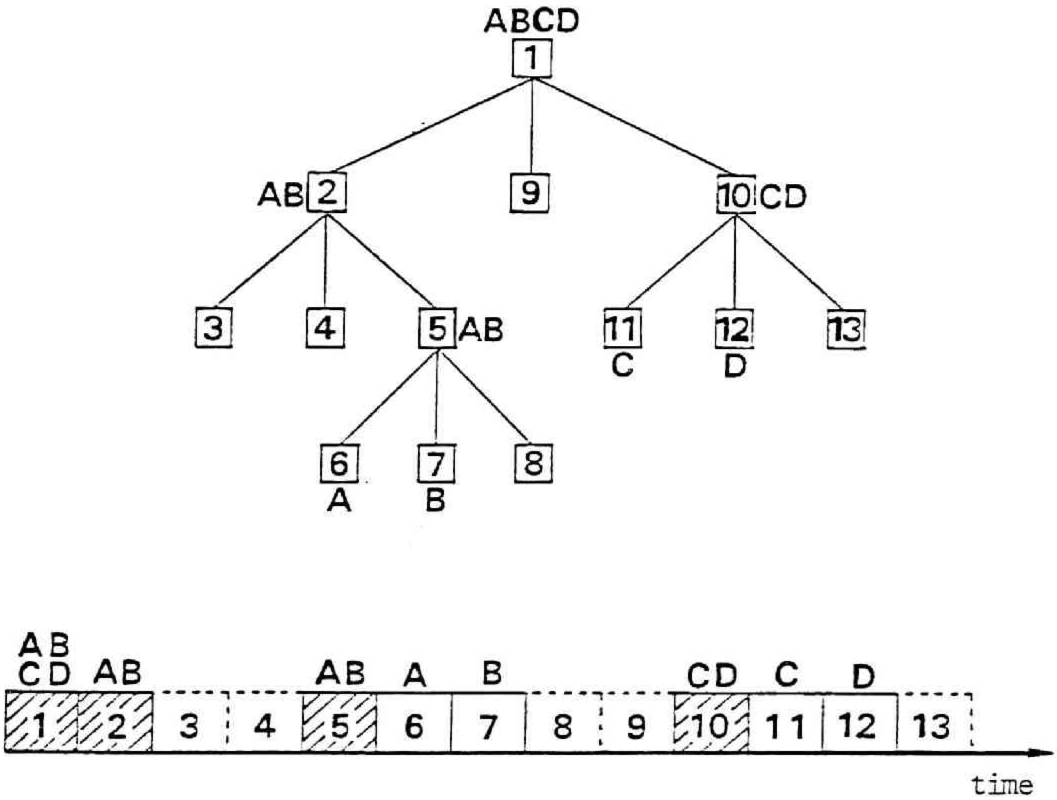


Figure 3.4 Example of collision resolution procedure in basic TA with Q=3

3.3 Analysis of Collision Resolution Time

In this section, we will only analyze a Q-ary BA TA/M-TF(m) to obtain its throughput characteristics, because it is clear from the discussion in section 3.2 that the Q-ary BA TA/M with binary feedback is equivalent to Q-ary BA TA/M with ternary feedback information where m=1. In this analysis, we assume that unit time is a large slot and that new packets arrivals in different (large) slots are independent and follow an identical Poisson process with rate λ (packets/large slot).

We first consider the CRT; i.e., the time required to resolve a collision, given that k number of packets are involved in the initial collision. Let $U_k(z)$ be a conditional probability generating function (pgf) of the CRT for a given collision multiplicity k. Namely,

$$U_k(z)=\sum_{i\geq 1} \text{Prob}[\text{CRT}=i\,|\,\text{Collision Multiplicity}=k]z^i \quad (k\geq 2)$$

Note that, since only one slot is used if k is zero or one, we have

$$U_0(z)=U_1(z)=z.$$

Since that a tree is divided into $s+m\times c$ number of subtrees in case of collision (see section 3.2 for the definition of s and c) and that new packets are not transmitted in a CRI, $U_k(z)$ is given by the following recursive equation:

$$U_k(z)=z\sum_{n_1+\dots+n_Q=k} p(k,n_1,\dots,n_Q)\{\prod_{i=1}^{Q} V_{n_i}(z)\}/Q^k, \qquad (k\geq 2) \tag{3.1}$$

where

$$V_n(z)=\begin{cases}\displaystyle\sum_{r_1+\dots+r_m=n} p(n,r_1,\dots,r_m)\{\prod_{j=1}^{m} U_{r_j}(z)\}/m^n & (n\geq 2)\\ 1\ (n=0),\ z\ (n=1).\end{cases} \tag{3.2}$$

In Eqs. (3.1) and (3.2), $p(k,n_1,\dots,n_Q)$ is the multinomial coefficient defined by

$$p(k,n_1,\dots,n_Q)=k!/(n_1!\dots n_Q!).$$

We denote the conditional average CRT of the Q-ary TA/M-TF(m) by M_k. Hence, from Eq. (3.1), we have

$$M_k = \frac{d}{dz}U_k(z)\Big|_{z=1}$$

$$=1+\sum_{n_1+\dots+n_Q=k} p(k,n_1,\dots,n_Q)\{\sum_{i=1}^{Q} v_{n_i}\}/Q^k, \quad (k\geq 2) \tag{3.3}$$

where v_n in Eq. (3.3) is defined by $\frac{d}{dz}V_n(z)|_{z=1}$ and thus takes the form

$$v_n=\begin{cases} \sum_{r_1+\dots+r_m=n} p(n,r_1,\dots,r_m)\{\sum_{j=1}^{m} M_{r_j}\}/m^n & (n\geq 2) \\ 0\ (n=0),\ 1\ (n=1). \end{cases} \tag{3.4}$$

We note that v_n means the average number of slots required to resolve a collision with multiplicity n if ternary feedback information is available in mini-slots. Then, the number of subtrees containing n packets when k packets are partitioned into Q subtrees is given by the following:

$$1/n!\sum_{i=1}^{Q}\frac{d^n}{dz_i^n}\sum_{n_1+\dots+n_Q=k} p(k,n_1,\dots,n_Q)z_1^{n_1}\dots z_Q^{n_Q}\Big|_{z_i=0,\ z_j=1\ (j\neq i)}$$

$$=Q\binom{k}{n}(Q-1)^{k-n} \qquad (n\geq 0).$$

Therefore, Eq. (3.3) can be rewritten as follows:

$$M_k=1+Q^{1-k}\sum_{n=0}^{k}\binom{k}{n}(Q-1)^{k-n}v_n. \tag{3.5}$$

By substituting Eq. (3.4) into Eq. (3.5) and using a way similar to that used to obtain Eq. (3.5), we can obtain the following recurrence equation for M_k:

$$M_k= 1-(m-1)(1-q)^{k-1}k-m(Q-1)(1-q)^{k-1}$$

$$+Qm\sum_{n=0}^{k}\binom{k}{n}(1-\alpha)^{k-n}\alpha^n M_n, \qquad (k\geq 2) \tag{3.6}$$

where $q=Q^{-1}$ and $\alpha=(Qm)^{-1}$.

Merakos et al. [MERA 83] derived a recurrence equation for M_k in a special case of Q=2 and m=2. Huang et al. [HUAN 85] obtained that for M_k for m=1. Now, we proceed to obtain the closed-form solution for M_k by the use of the generating function method presented in [HOFR 84].

Now, we define

$$M(z)=\sum_{k\geq 0} M_k z^k/k!. \tag{3.7}$$

By multiplying both sides of Eq. (3.6) by $z^k/k!$ and summing up over $k\geq 0$, we have

$$M(z)e^{-z}-QmM(\alpha z)e^{-z} = 1-ze^{-z}-\{Qm+(m-1)z\}e^{-qz}.$$

Definig $M^*(z)$ as

$$M^*(z)=e^{-z}M(z), \tag{3.8}$$

then we have

$$M^*(z)-QmM^*(\alpha z) = 1-ze^{-z}-\{Qm+(m-1)z\}e^{-qz}.$$

We shall write $M^*(z)$ as $M^*(z)=\sum_{n\geq 0} M^*_n z^n$ in terms of M^*_n. Furthermore, equating the coefficients of z^n in the above equation, M^*_n is given by

$$M^*_n=\begin{cases} 1 \quad (n=0), \qquad 0 \quad (n=1) \\ [\{1+q^{n-1}(m-1)\}n-\alpha^{-1}q^n](-1)^n/\{n!(1-\alpha^{n-1})\} \quad (n\geq 2). \end{cases} \tag{3.9}$$

From the definitions (3.7) and (3.8), we have the following relationship between M_k and M^*_n:

$$M_k=k!\sum_{n=0}^{k} M^*_n/(k-n)!.$$

Finally, using Eq. (3.9), we can obtain the following closed-form solution for M_k:

$$M_k=1+k+\sum_{n=2}^{k}(-1)^n\binom{k}{n}\{n(m^n-m^{n-1}+1)-m^n\}/((Qm)^{n-1}-1). \quad (k\geq 2) \tag{3.10}$$

Here, we denote the average conditional CRT of the Q-ary basic TA and the TA/M-BF by T_k and B_k, respectively. We can obtain the closed-form solutions for T_k and B_k corresponding to Eq. (3.10) in a similar way:

$$T_k=1+Q+Q\sum_{n=2}^{k}(-1)^n(n-1)\binom{k}{n}/(Q^{n-1}-1), \tag{3.11}$$

$$B_k=1+k+\sum_{n=2}^{k}(-1)^n(n-1)\binom{k}{n}/(Q^{n-1}-1). \tag{3.12}$$

The former, Eq. (3.11), has already been obtained in [MATH 85, MURO 85]. Note that we can obtain Eq. (3.12) easily by letting m=1 in Eq. (3.10). It follows from Eqs. (3.11) and (3.12) that the next relationship holds between T_k and B_k:

$$B_k=k-1/Q+T_k/Q. \tag{3.13}$$

Furthermore, we note that (T_k-B_k) represents the average number of empty slots during a CRI in the Q-ary basic TA because the Q-ary TA/M-BF has been designed to improve performance by completely eliminating empty slots as mentioned earlier in section 3.2.

In the next section, we will evaluate the <u>maximum throughput</u> by means of the conditional average CRT, M_k.

3.4. Throughput analysis

From theorem of Pakes [PAKE 69], a TA/M-TF(m) is stable if an input rate is smaller than $S_{inf}(Q,m)$ defined as

$$S_{inf}(Q,m) = \liminf_{k \to \infty} \frac{k}{M_k}. \tag{3.14}$$

On the other hand, a TA/M-TF(m) is unstable if an input rate is larger than $S_{sup}(Q,m)$ given by

$$S_{sup}(Q,m) = \limsup_{k \to \infty} \frac{k}{M_k}. \tag{3.15}$$

$S_{inf}(Q,m)$ will be referred to as the <u>stable maximum throughput</u>.

Defining $A(k,Q,m)$ by

$$A(k,Q,m) = \sum_{n=2}^{k} (-1)^n \binom{k}{n} \{n(m^n - m^{n-1} + 1) - m^n\} / ((Qm)^{n-1} - 1),$$

from Eq.(3.10), we have

$$M_k = 1 + k + A(k,Q,m),$$

and

$$\lim_{k \to \infty} k/M_k = \lim_{k \to \infty} k/(k + A(k,Q,m) + 1)$$

$$= \lim_{k \to \infty} \frac{1}{1 + (A(k,Q,m)/k)}. \tag{3.16}$$

In the following, in order to obtain $S_{inf}(Q,m)$ and $S_{sup}(Q,m)$, we will analyze the asymptotic behavior of $A(k,Q,m)/k$ when k approaches infinity.

First, we shall rewrite $A(k,Q,m)$ as follows:

$$A(k,Q,m) = \sum_{r \geq 1} \left[\sum_{n=2}^{k} (-1)^n \binom{k}{n} \{n(m^n - m^{n-1} + 1) - m^n\} (Qm)^{-(n-1)r} \right]$$

$$= \sum_{r \geq 1} \left[\sum_{n=2}^{k} (-1)^n \binom{k}{n} \{n(m^n - m^{n-1}) + n - m^n\} (Qm)^{-(n-1)r} \right]$$

$$= \sum_{r \geq 1} \left[(m-1)\{-k(1 - m(Qm)^{-r})^k + k\} + \{-k(1 - (Qm)^{-r})^k + k\} \right.$$

$$\left. + (-1)(Qm)^r \{(1 - mQ^{-r}m^{-r})^k - (1 - kmQ^{-r}m^{-r})\} \right].$$

Further, let us rewrite $A(k,Q,m)$ as follows:

$$A(k,Q,m)=k\sum_{r\geq 1} u_k(k(Qm)^{-r}),$$

where

$$u_k(x)=(m-1)\{-(1-mx/k)^k+1\}-(1-x/k)^k+1-\{(1-mx/k)^k-(1-mx)\}/x. \qquad (3.17)$$

Thus, we get

$$\lim_{k\to\infty} A(k,Q,m)/k = \lim_{k\to\infty}\sum_{r\geq 1} u_k(k(Qm)^{-r}). \qquad (3.18)$$

Now, we will give the following lemma (see Appendix 3-A for its proof).

<u>Lemma 3.1.</u> Let $g_s(x)$ $(s=1,2,\ldots)$ and $G(x)$ be continuous functions defined on $[0,+\infty)$ and satisfy the following conditions:

(C-1) As $s\to+\infty$, $g_s(x)$ uniformly converges to $G(x)$ on any bounded interval in $[0,+\infty)$.

(C-2) There exist some positive constants C, a, b such that

$|g_s(x)|\leq C/x^a$ and $|G(x)|\leq C/x^a$ for $x\geq 1$,

$|g_s(x)|\leq Cx^b$ and $|G(x)|\leq Cx^b$ for $0\leq x\leq 1$.

Then, for any constant $d>1$, it holds that

$$\lim_{s\to\infty}\left|\sum_{n\geq 1} g_s(sd^{-n})-\sum_{p=-\infty}^{\infty} G(d^{(\ln(s)/\ln(d))-\lfloor \ln(s)/\ln(d)\rfloor^{*}-p})\right|=0. \qquad \square$$

Here, we note the following; i.e., by <u>Poisson's summation formula</u> [HENR 77], it is shown that

$$\sum_{p=-\infty}^{\infty} G(d^{(\ln(s)/\ln(d))-\lfloor \ln(s)/\ln(d)\rfloor-p})$$
$$=\sum_{q=-\infty}^{\infty}(1/\ln(d))d^{(2\pi/\ln(d))\{(\ln(s)/\ln(d))-\lfloor \ln(s)/\ln(d)\rfloor\}qi}\cdot M(G,-(2\pi/\ln(d))qi),$$

where

$$M(G,t)=\int_0^\infty G(z)z^{t-1}dz$$ (<u>Mellin transformation</u> [HENR 77]).

---

* $\lfloor x\rfloor$ denotes the greatest integer equal to or less than x.

In order to apply Lemma 3.1 to $A(k,Q,m)/k$, we shall prove the following lemma. Note that $u_k(x)$ given by Eq.(3.17) is defined for x such that $0<x\leq k/m$ because $x=k(Qm)^{-r}$ $(Qm>1,\ r\geq 1)$.

Lemma 3.2. For a given positive integer m, we shall consider a function $f_s(x)$ $(s=1,2,\ldots)$ and $F(x)$ given by

$$f_s(x)=\begin{cases} 0 & x=0 \\ u_s(x) & 0<x\leq s/m \\ u_s(s/m)(s/m)/x & x\geq s/m, \end{cases}$$

and

$$F(x)=(m-1)(-e^{-mx}+1)-e^{-x}+1-(e^{-mx}-1+mx)/x.$$

Then, $f_s(x)$ and $F(x)$ are continuous functions defined on $[0,+\infty)$ and satisfy the conditions (C-1) and (C-2) of Lemma 3.1.

Proof. It is clear that $f_s(x)$ and $F(x)$ are continuous functions. Since, as $s\to+\infty$, $(1-y/s)^s$ uniformly converges to e^{-y} on any bounded interval in $[0,+\infty)$, it is shown that $f_s(x)$ uniformly converges to $F(x)$ on any bounded interval in $[0,+\infty)$ as $s\to+\infty$. We can easily show the same property of $f_s(x)$ and $F(x)$ is satisfied when $x=0$ and $x\geq s/m$. Thus, $f_s(x)$ and $F(x)$ satisfy the condition (C-1) of Lemma 3.1.

Now, we shall show that $f_s(x)$ satisfies the condition (C-2) of Lemma 3.1. First assume that $x\geq 1$. If $s/m\leq 1$, then it is easily shown that

$$f_s(x)=u_s(s/m)(s/m)/x\leq C_1/x \quad (C_1>0,\ \text{constant}).$$

Thus, assume that $s/m\geq 1$. For $x\geq s/m$ (≥ 1), the above inequality also holds. In the case that $1\leq x\leq s/m$, it holds that

$$1\geq mx/s\geq m/s \ (>0).$$

By the fact that

$$y(1-y)^t\leq 1/(t+1) \quad (0\leq y\leq 1,\ t\geq 0),$$

we can easily show that

$(1-mx/s)^s \leq 1/(mx)$,

$(1-x/s)^s \leq 1/x$.

Also, considering the fact

$$f_s(x)=(m-1)\{-(1-mx/s)^s\}-(1-x/s)^s+\{(1-mx/s)^s-1\}/x,$$

it is shown that there exists some positive integer C_2 such that

$|f_s(x)| \leq C_2/x$ for $x \geq 1$.

Next, we assume that $0 \leq x \leq 1$. $f_s(x)$ is written as follows:

$$f_s(x)=-(m-1)\sum_{i=1}^{s}\binom{s}{i}(-mx/s)^i-\sum_{i=1}^{s}\binom{s}{i}(-x/s)^i-\sum_{i=2}^{s}\binom{s}{i}(-mx/s)^i/x$$

Thus, it holds that

$|f_s(x)| \leq C_3 x$ for $0 \leq x \leq 1$.

Consequently, it is shown that $f_s(x)$ satisfies the condition (C-2) of Lemma 3.1.

Considering $\lim_{s\to\infty} f_s(x)=F(x)$, we can easily prove that $F(x)$ satisfies the condition (C-2) of Lemma 3.1. □

From Lemma 3.1 and Lemma 3.2, we easily obtain the following theorem.

Theorem 3.1.

$$\lim_{k\to\infty} \left|A(k,Q,m)/k-\sum_{p=-\infty}^{\infty} F((Qm)^{(\ln(k)/\ln(Qm))-\lfloor \ln(k)/\ln(Qm)\rfloor-p})\right|=0.$$

Proof. Since $x=k(Qm)^{-n}$ $(n \geq 1)$ leads to $x \leq k/m$, we have

$$\sum_{n\geq 1} f_k(k(Qm)^{-n})=\sum_{n\geq 1} u_k(k(Qm)^{-n}).$$

Thus, the theorem follows from Eq.(3.18), Lemma 3.1 and Lemma 3.2. □

Let α denote $(\ln(k)/\ln(Qm))-\lfloor \ln(k)/\ln(Qm)\rfloor$; thus, $0 \leq \alpha < 1$. Defining the following function:

$$g(\alpha)=\sum_{p=-\infty}^{\infty} F((Qm)^{\alpha-p}),$$

we have from Theorem 3.1

$$\liminf A(k,Q,m)/k = \inf_{0\leq\alpha<1} g(\alpha), \tag{3.17}$$

$$\limsup A(k,Q,m)/k = \sup_{0\leq\alpha<1} g(\alpha). \tag{3.18}$$

Therefore, from Eqs.(3.14), (3.15) and (3.16), we have

$$S_{inf}(Q,m)=1/(1+\sup g(\alpha)),$$

$$S_{sup}(Q,m)=1/(1+\inf g(\alpha)).$$

We can numerically obtain these values of $\inf g(\alpha)$ and $\sup g(\alpha)$ (see Appendix 3-B for numerical computations). Figure 3.5 illustrates $g(\alpha)$ with Q=16 and m=1 as a function of α. As shown in this figure, $g(\alpha)$ has two extrema, which are $\sup_{0\leq\alpha<1} g(\alpha)$ and $\inf_{0\leq\alpha<1} g(\alpha)$.

Recently, Mathys and Flajolet [MATH 85] obtained an explicit expression for the stable maximum throughput of the basic tree algorithm (without mini-slots). In what follows, we will obtain a similar expression in the TA/M.

Corollary 3.1.

$$\lim_{k\to\infty} \Big| A(k,Q,m)/k-[(1/\ln(Qm)) \sum_{p=-\infty}^{\infty} \{(m^{(2\pi p i/\ln(Qm))}-1)/(-2\pi p i/\ln(Qm))$$

$$+(m^{1+(2\pi p i/\ln(Qm))})/(1+(2\pi p i/\ln(Qm))\}$$

$$\cdot\Gamma(1-(2\pi p i/\ln(Qm)))e^{2\pi p i\{(\ln(k)/\ln(Qm))-\lfloor\ln(k)/\ln(Qm)\rfloor\}}] \Big|=0,$$

where $\Gamma(x)$ is a gamma function.

Proof. see Appendix 3-C. □

We obtain as a special case (i.e., m=1) the following corollary.

Corollary 3.2.

$$\lim_{k\to\infty} \Big| A(k,Q,1)/k-[(1/\ln(Q)) \sum_{p=-\infty}^{\infty} \{(1/(1+2\pi p i/\ln(Q))\}\Gamma(1-(2\pi p i/\ln(Q)))$$

$$\cdot e^{2\pi p i\{(\ln(k)/\ln(Q))-\lfloor\ln(k)/\ln(Q)\rfloor\}}] \Big|=0$$

□

This corollary is the same result obtained by Mathys and Flajolet [MATH 85] through the Mellin transformation method.

From Corollaries 3.1 and 3.2, it becomes clear that as k approaches infinity, the function $A(k,Q,m)/k$ does not converge to any function which is independent of k, but some function which oscillates around $(m-\ln(m))/\ln(Qm)$, which is the term corresponding to p=0 of the summation in Corollary 3.1. We define $\bar{A}(Q,m)$ as

$$\bar{A}(Q,m)=(m-\ln(m))/\ln(Qm). \tag{3.21}$$

Equation (3.21) is an explicit expression concerned with throughput, but is not an critical value of maximum throughput. To compare with $S_{inf}(Q,m)$ and $S_{sup}(Q,m)$ later, we define the following quantity:

$$S^*(Q,m)=1/(1+\bar{A}(Q,m))$$
$$=\ln(Qm)/(m+\ln(Q)). \tag{3.22}$$

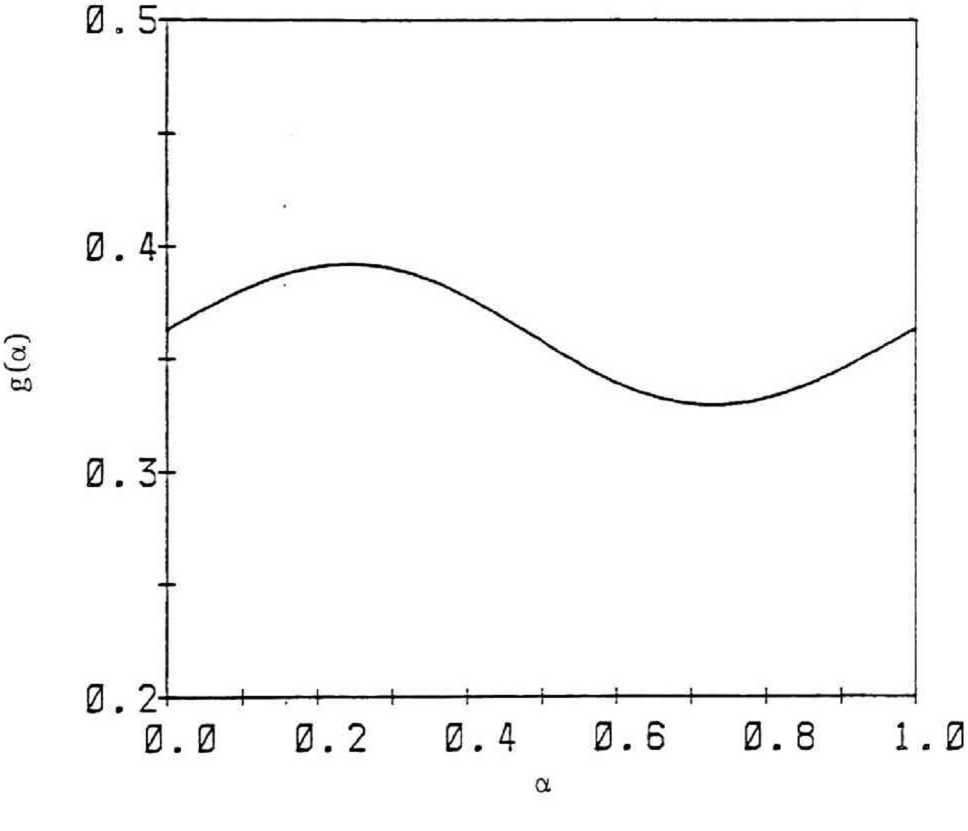


Figure 3.5 $g(\alpha)$ versus α

Table 3.1

$S_{inf}(Q,1)$, $S^*(Q,1)$ and $S_{sup}(Q,1)$

Q	$S_{inf}(Q,1)$	$S^*(Q,1)$	$S_{sup}(Q,1)$
2	0.40938	0.40938	0.40938
3	0.52343	0.52349	0.52356
4	0.58049	0.58094	0.58139
5	0.61552	0.61678	0.61804
6	0.63942	0.64180	0.64422
7	0.65684	0.66055	0.66433
8	0.67011	0.67527	0.68054
9	0.68059	0.68723	0.69406
10	0.68906	0.69721	0.70562
11	0.69606	0.70570	0.71568
12	0.70195	0.71305	0.72457
13	0.70696	0.71949	0.73251
14	0.71129	0.72520	0.73968
15	0.71506	0.73032	0.74620
16	0.71838	0.73493	0.75218
17	0.72132	0.73912	0.75768
18	0.72394	0.74296	0.76278
19	0.72629	0.74648	0.76752
20	0.72842	0.74973	0.77195
21	0.73035	0.75275	0.77609
22	0.73211	0.75556	0.77999
23	0.73372	0.75819	0.78367
24	0.73520	0.76065	0.78714
25	0.73656	0.76297	0.79043

Table 3.2

$S_{inf}(Q,2)$, $S^*(Q,2)$ and $S_{sup}(Q,2)$

Q	$S_{inf}(Q,2)$	$S^*(Q,2)$	$S_{sup}(Q,2)$
2	0.51459	0.51475	0.51491
3	0.57792	0.57825	0.57855
4	0.61222	0.61408	0.61590
5	0.63394	0.63793	0.64192
6	0.64897	0.65534	0.66181
7	0.65999	0.66881	0.67786
8	0.66843	0.67965	0.69128
9	0.67510	0.68864	0.70280
10	0.68050	0.69626	0.71285
11	0.68497	0.70285	0.72177
12	0.68872	0.70861	0.72975
13	0.69193	0.71372	0.73697
14	0.69469	0.71829	0.74356
15	0.69709	0.72242	0.74959
16	0.69921	0.72618	0.75516
17	0.70108	0.72961	0.76032
18	0.70276	0.73277	0.76513
19	0.70426	0.73569	0.76962
20	0.70561	0.73841	0.77383
21	0.70684	0.74094	0.77778
22	0.70796	0.74330	0.78152
23	0.70899	0.74553	0.78504
24	0.70993	0.74762	0.78839
25	0.71080	0.74959	0.79156

Table 3.3

$S_{inf}(Q,3)$, $S^*(Q,3)$ and $S_{sup}(Q,3)$

Q	$S_{inf}(Q,3)$	$S^*(Q,3)$	$S_{sup}(Q,3)$
2	0.48265	0.48516	0.48769
3	0.53123	0.53609	0.54124
4	0.55958	0.56652	0.57405
5	0.57797	0.58750	0.59740
6	0.59071	0.60320	0.61547
7	0.60004	0.61556	0.63024
8	0.60717	0.62567	0.64277
9	0.61281	0.63415	0.65371
10	0.61737	0.64142	0.66345
11	0.62114	0.64775	0.67224
12	0.62431	0.65334	0.68026
13	0.62701	0.65833	0.68763
14	0.62934	0.66282	0.69443
15	0.63137	0.66689	0.70075
16	0.63315	0.67062	0.70663
17	0.63474	0.67404	0.71214
18	0.63615	0.67720	0.71731
19	0.63741	0.68014	0.72217
20	0.63856	0.68288	0.72676
21	0.63960	0.68544	0.73110
22	0.64054	0.68784	0.73521
23	0.64141	0.69010	0.73912
24	0.64220	0.69224	0.74284
25	0.64293	0.69426	0.74639

3.5 Numerical and simulation results

In Tables 3.1, 3.2 and 3.3, we give values of $S_{inf}(Q,m)$, $\bar{S}(Q,m)$ and $S_{sup}(Q,m)$ for m=1, 2 and 3, respectively. $S_{inf}(2,2)$ is not a new result, but was first obtained by Merakos and Kazakos [MERA 83]. Figures 3.6 and 3.7 illustrate the same cases as in Table 3.1 and 3.2. It is seen that difference between $S_{inf}(Q,m)$ and $S_{sup}(Q,m)$ is smaller than 0.01 for Q such that $Q \leq 7$ when m=1; for $Q \leq 5$ when m=2; and only for Q=2 when m=3; The difference becomes large as Q becomes large. In all cases except such several ones as the difference is rather small, $\bar{S}(Q,m)$ is not used to measure the performance of the algorithms. Instead, we should numerically obtain $S_{inf}(Q,m)$. Furthermore, by comparing Tables 3.1, 3.2 and 3.3, we see that, when $Q \leq 7$, $S_{inf}(Q,2)$ is the greatest, and, when $Q \geq 8$, $S_{inf}(Q,1)$ takes the greatest value.

In addition, by the use of $S_{inf}(Q,m)$, we consider the effect of the length of mini-slot, denoted by h, on the maximum throughput. Letting $\tilde{S}(Q,m)$ denote the stable maximum throughput in this case, we have

$$\tilde{S}(Q,m) = S_{inf}(Q,m)/(1+hQ).$$

As shown in Tables 3.1, 3.2 and 3.3, $S_{inf}(Q,m)$ is an increasing function of Q; on the other hand, $1/(1+hQ)$ is a decreasing one of Q. Thus, $\tilde{S}(Q,m)$ takes the maximum value for a given value of h. Table 3.4 shows $\tilde{S}(Q,1)$ as a function of Q for h=0.001. When h=0.001, an optimum value of Q is equal to 34. For example, in a system with 2000 bit data packets and 2 bit long mini-slot, h is equal to 0.001. Table 3.4 shows that such a system attains the stable maximum throughput 0.7208 when m=1.

For several values of h, optimum values of Q and the corresponding values of both $\tilde{S}(Q,1)$ and $\tilde{S}(Q,2)$ are given in Table 3.5. By comparing $\tilde{S}(Q,1)$ and $\tilde{S}(Q,2)$, we see that, for small values of h such as $h \leq 0.017$, m=1 gives the better performance, and that, otherwise (i.e., for

$h \geq 0.018$), m=2 gives the better performance. However, even when h=0.025, the difference between $\bar{S}(Q,1)$ and $\bar{S}(Q,2)$ is about 0.005.

It is small values of h that give the TA/M an advantage over algorithms without mini-slots. For such small values of h, m=1 provides the best performance. Furthermore, the TA/M-TF(1), i.e., the TA/M-BF, needs only something/nothing binary feedback in a mini-slot, and hence it is easier to implement than other TA/M-TF(m)s ($m \geq 2$). Therefore, we conclude that the TA/M-BF is an excellent algorithm.

Next, we show the average transmission delay for the TA/M-BF and the TA/M-TF(2) in Figs. 3.7 and 3.8, respectively. Note that both $S_{inf}(Q,m)$ and $S_{sup}(Q,m)$ do not refer to the achievable maximum throughput in the equilibrium. We see that the achievable maximum throughput is between $S_{inf}(Q,m)$ and $S_{sup}(Q,m)$ for such values of Q as given in these figures.

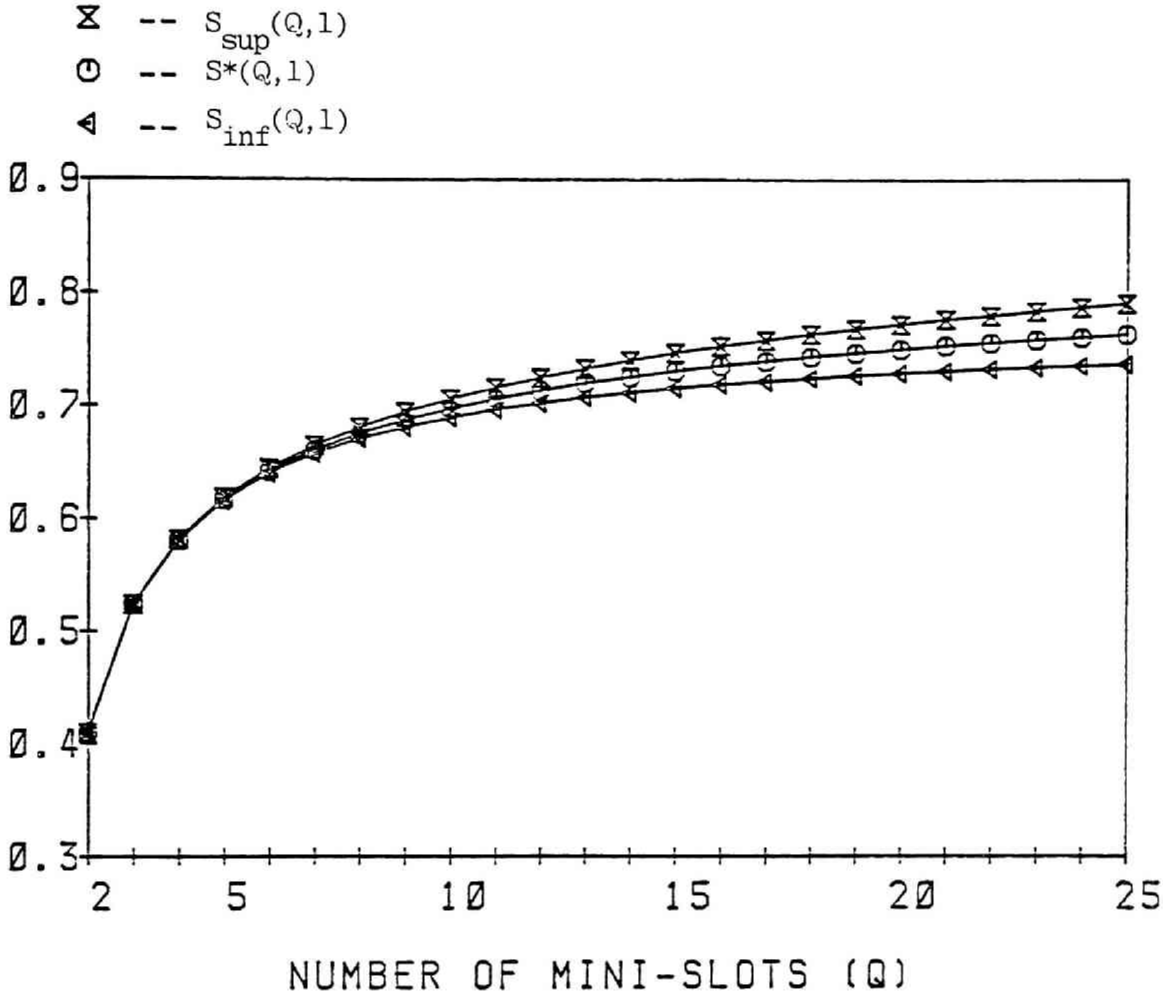


Figure 3.6 Comparison of $S_{inf}(Q,1)$, $S^*(Q,1)$ and $S_{sup}(Q,1)$

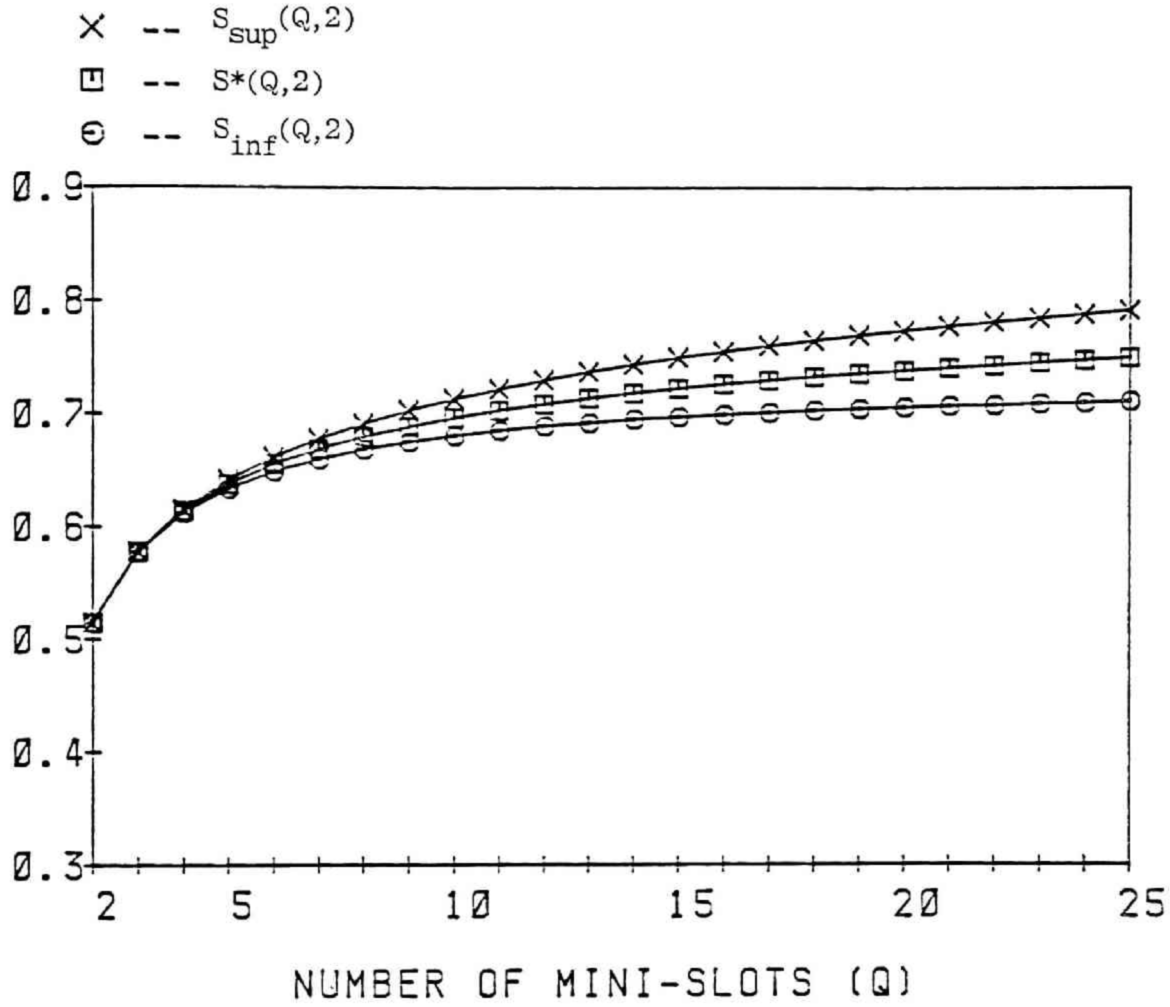


Figure 3.7 Comparison of $S_{inf}(Q,2)$, $S^*(Q,2)$ and $S_{sup}(Q,2)$

Table 3.4

$\bar{S}(Q,1)$ in the case of h=0.001

Q	$\bar{S}(Q,1)$
2	0.408563
3	0.521864
4	0.578177
5	0.612458
6	0.635606
7	0.652274
8	0.664792
9	0.674519
10	0.682238
11	0.688487
12	0.693626
13	0.697887
14	0.701469
15	0.704493
16	0.707067
17	0.709263
18	0.711139
19	0.712748
20	0.714137
21	0.715328
22	0.716350
23	0.717224
24	0.717969
25	0.718595
26	0.719123
27	0.719562
28	0.719922
29	0.720204
30	0.720427
31	0.720592
32	0.720698
33	0.720765
34	0.720783
35	0.720754
36	0.720695
37	0.720608
38	0.720482
39	0.720327
40	0.720154

Table 3.5

Optimum values of Q and $\bar{S}(Q_{opt},m)$

h	Q_{opt}	$\bar{S}(Q_{opt},1)$	Q_{opt}	$\bar{S}(Q_{opt},2)$
0.001	34	0.720783	27	0.693622
0.002	24	0.701527	19	0.678478
0.003	20	0.687189	16	0.667185
0.004	17	0.675393	14	0.657850
0.005	15	0.665172	12	0.649736
0.006	14	0.656172	11	0.642561
0.007	13	0.647993	11	0.635998
0.008	12	0.640465	10	0.630093
0.009	12	0.633529	9	0.624514
0.010	11	0.627081	9	0.619358
0.011	11	0.620928	8	0.614366
0.012	10	0.615232	8	0.609881
0.013	10	0.609788	8	0.605462
0.014	10	0.604439	8	0.601106
0.015	9	0.599639	7	0.597276
0.016	9	0.594921	7	0.593516
0.017	9	0.590278	7	0.589803
0.018	8	0.585760	7	0.586137
0.019	8	0.581693	6	0.582558
0.020	8	0.577681	6	0.579437
0.021	8	0.573724	6	0.576350
0.022	8	0.569821	6	0.573295
0.023	8	0.565971	6	0.570272
0.024	7	0.562363	6	0.567281
0.025	7	0.559013	6	0.564322

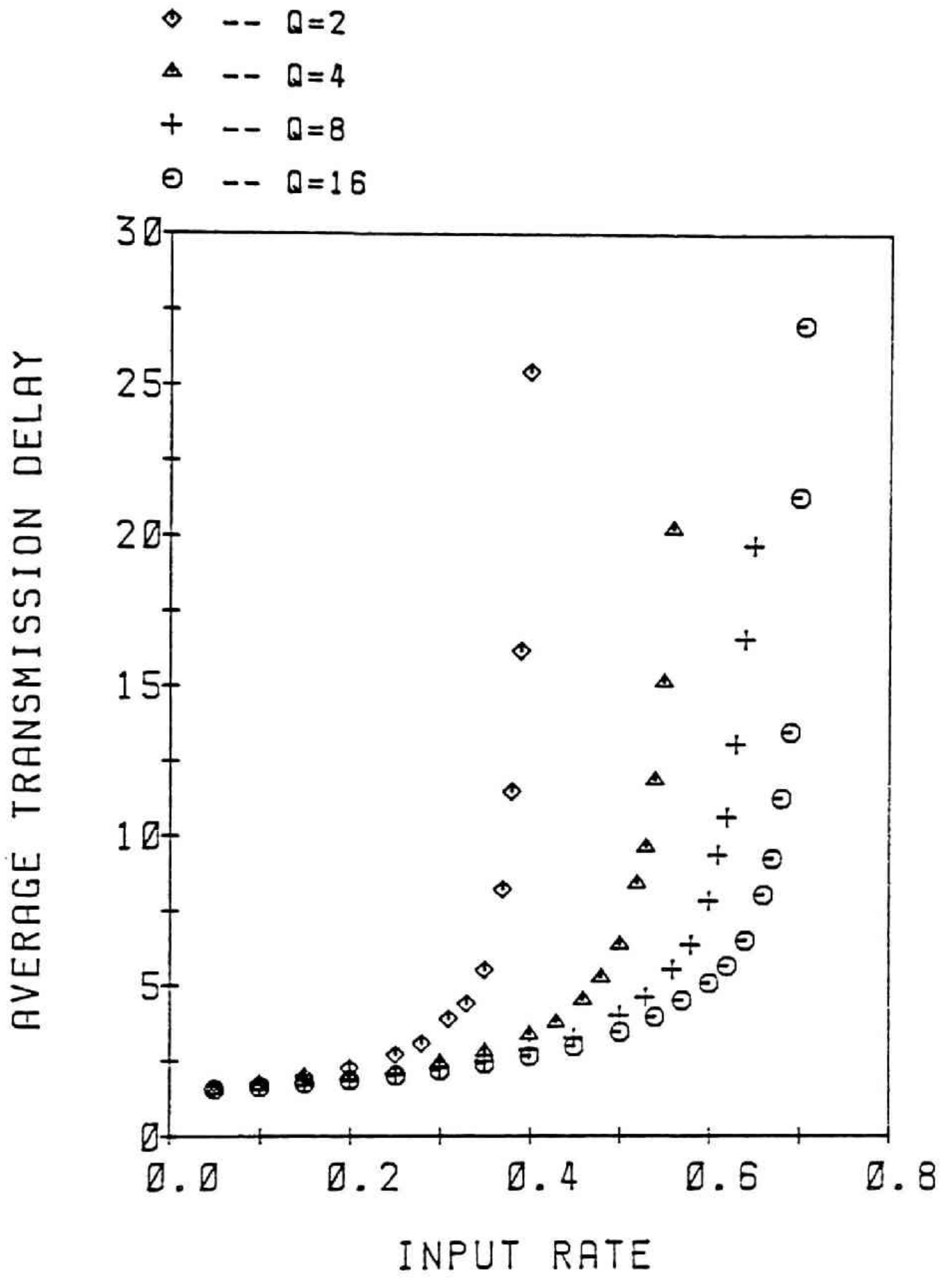


Figure 3.8 Average transmission delay of a TA/M-BF (Q=2, 4, 8, 16)

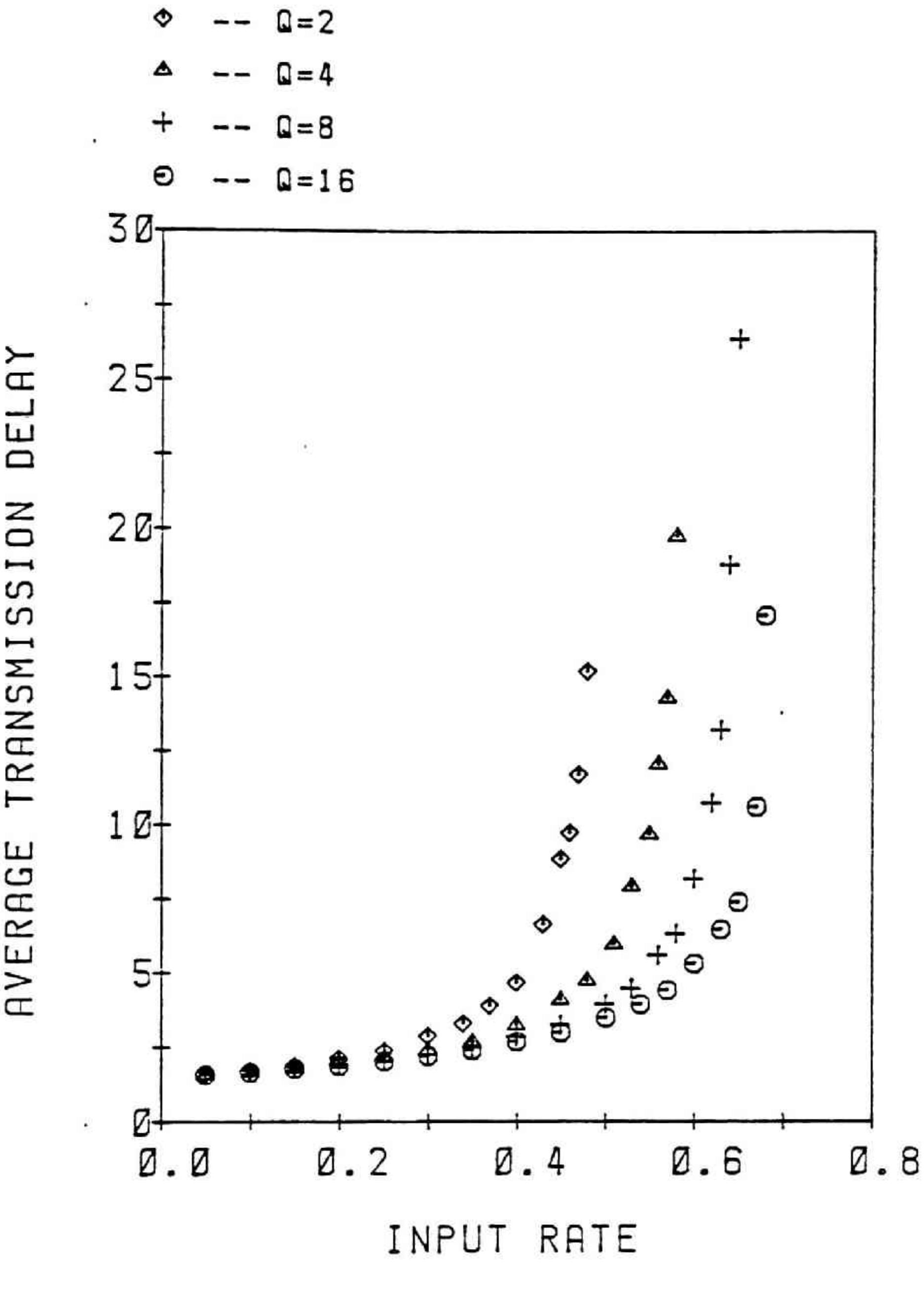


Figure 3.9 Average transmission delay of a TA/M-TF(2)
(Q=2, 4, 8, 16)

Table 3.6

$S_{inf}(Q)$, $S^*(Q)$ and $S_{sup}(Q)$

in the basic Q-ary tree algorithm

Q	$S_{inf}(Q)$	$S^*(Q)$	$S_{sup}(Q)$
2	0.34657	0.34657	0.34657
3	0.36611	0.36620	0.36630
4	0.34593	0.34657	0.34722
5	0.32018	0.32189	0.32362

3.6 Concluding remarks

In this chapter, we obtained the stable maximum throughput of the blocked access TA/M-TF(m) and TA/M-BF. We gave a way to obtain their stable maximum throughput numerically. Our approach will be easily applied to other blocked access tree type algorithms if their closed form expression for the averarage CRT is obtained. Furthermore, we obtained an optimum value of Q maximizing the stable maximum throughput for a given mini-slot length (denoted by h).

From numerical results, it has become clear that the TA/M-TF(1), i.e., TA/M-BF, is superior to other TA/M-TF(m)s ($m \geq 2$) for small values of h (i.e., $h \leq 0.017$). It is small values of h that give the TA/M an advantage over algorithms without mini-slots. Furthermore, the TA/M-TF(1), i.e., the TA/M-BF, needs only something/nothing binary feedback in a mini-slot, and hence it is easier to implement than other TA/M-TF(m)s ($m \geq 2$). Therefore, we conclude that the TA/M-BF is an excellent algorithm.

In other words, since larger feedback information can provide better performance, this suggests that a more complicated procedure is required to use ternary feedback information more sufficiently in retransmission when Q is large.

Appendix 3-A **Proof of Lemma 3.1**

Lemma 3.1. Let $g_s(x)$ $(s=1,2,\ldots)$ and $G(x)$ be continuous functions defined on $[0,+\infty)$ and satisfy the following conditions:

(C-1) As $s\to+\infty$, $g_s(x)$ uniformly converges to $G(x)$ on any bounded interval in $[0,+\infty)$.

(C-2) There exist some positive constants C, a, b such that

$$|g_s(x)|\leq C/x^a \text{ and } |G(x)|\leq C/x^a \text{ for } x\geq 1, \tag{3A.1}$$

$$|g_s(x)|\leq Cx^b \text{ and } |G(x)|\leq Cx^b \text{ for } 0\leq x\leq 1. \tag{3A.2}$$

Then, for any constant $d>1$, it holds that

$$\lim_{s\to\infty}\Big|\sum_{n\geq 1} g_s(sd^{-n})-\sum_{p=-\infty}^{\infty} G(d^{(\ln(s)/\ln(d))-\lfloor \ln(s)/\ln(d)\rfloor-p})\Big|=0.$$

Proof. We will broadly divide the summation $\sum_{n\geq 1} g_s(sd^{-n})$ (or $G(sd^{-n})$) into the following three parts:

$$\sum_{n\geq 1} g_s(sd^{-n})=\sum_{sd^{-n}\leq\frac{1}{N}} g_s(sd^{-n})+\sum_{\frac{1}{N}<sd^{-n}<N} g_s(sd^{-n})+\sum_{sd^{-n}\geq N} g_s(sd^{-n}).$$

Letting N_ε be $\frac{\ln(N)}{\ln(d)}$, we rewrite the above summation as

$$\sum_{n\geq 1} g_s(sd^{-n})=\sum_{|n-\lfloor\frac{\ln(s)}{\ln(d)}\rfloor|<N_\varepsilon} g_s(sd^{-n})$$

$$+\sum_{n\leq\lfloor\frac{\ln(s)}{\ln(d)}\rfloor-N_\varepsilon} g_s(sd^{-n})+\sum_{n\geq\lfloor\frac{\ln(s)}{\ln(d)}\rfloor+N_\varepsilon} g_s(sd^{-n}).$$

First, for any n satisfying $|n-\lfloor \ln(s)/\ln(d)\rfloor|<N_\varepsilon$, it holds that

$$\begin{aligned}d^{-N_\varepsilon}&\leq d^{(\ln(s)/\ln(d))-\lfloor\ln(s)/\ln(d)\rfloor}\\&<sd^{-n}\\&<d^{(\ln(s)/\ln(d))-\lfloor\ln(s)/\ln(d)\rfloor+N_\varepsilon}\\&\leq d^{1+N_\varepsilon}.\end{aligned}$$

Thus, sd^{-n} is in a bounded interval $(d^{-N_\varepsilon},d^{1+N_\varepsilon})$ $(\epsilon[0,+\infty))$ which is independent of s. From the condition (C-1), we can show that there exists a constant $K_\varepsilon>0$ such that

$$|g_s(sd^{-n})-G(sd^{-n})|<\varepsilon/(2N_\varepsilon) \tag{3A.3}$$

is satisfied for $s\geq K_\varepsilon$. Furthermore, from (3A.3), it holds for $s\geq K_\varepsilon$ that

$$\left|\sum_{|n-\lfloor \ln(s)/\ln(d)\rfloor|<N_\varepsilon} g_s(sd^{-n})-\sum_{|n-\lfloor \ln(s)/\ln(d)\rfloor|<N_\varepsilon} G(sd^{-n})\right|<\varepsilon. \tag{3A.4}$$

Second, from the condition (C-2), we shall show that for any $\varepsilon>0$, there exists a constant $N_\varepsilon(\geq 1)$ such that

$$\sum_{n\leq\lfloor \ln(s)/\ln(d)\rfloor-N_\varepsilon} |g_s(sd^{-n})|<\varepsilon, \tag{3A.5}$$

$$\sum_{n\leq\lfloor \ln(s)/\ln(d)\rfloor-N_\varepsilon} |G(sd^{-n})|<\varepsilon, \tag{3A.6}$$

$$\sum_{n\geq\lfloor \ln(s)/\ln(d)\rfloor+N_\varepsilon} |g_s(sd^{-n})|<\varepsilon, \tag{3A.7}$$

$$\sum_{n\geq\lfloor \ln(s)/\ln(d)\rfloor+N_\varepsilon} |G(sd^{-n})|<\varepsilon. \tag{3A.8}$$

We show now that (3A.5) is satisfied for a constant $N'_\varepsilon(\geq 1)$. Since $s=d^{(\ln(s)/\ln(d))}$ and $n\leq\lfloor \ln(s)/\ln(d)\rfloor-1$, it holds that

$$\begin{aligned}
sd^{-n}&=d^{(\ln(s)/\ln(d))}d^{-n}\\
&=d^{(\ln(s)/\ln(d))-n}\\
&\geq d^{\lfloor \ln(s)/\ln(d)\rfloor-n}\\
&\geq d\\
&\geq 1.
\end{aligned}$$

From (3A.1) of the condition (C-2), we can assume that

$$\begin{aligned}
|g_s(sd^{-n})|&\leq C(1/sd^{-n})^a\\
&=Cs^{-a}d^{na}.
\end{aligned}$$

Hence,

$$\sum_{n\leq\lfloor \ln(s)/\ln(d)\rfloor-N'_\varepsilon} |g_s(sd^{-n})| \leq \sum_{n\leq\lfloor \ln(s)/\ln(d)\rfloor-N'_\varepsilon} Cs^{-a}d^{na}$$

$$=Cs^{-a}\{d^{a(\lfloor \ln(s)/\ln(d)\rfloor-N'_\varepsilon)}-1\}/(d^a-1)$$

$$\leq C\{d^{(\lfloor \ln(s)/\ln(d)\rfloor-N'_\varepsilon)}s^{-1}\}^a/(d^a-1)$$

$$=C\{d^{\lfloor \ln(s)/\ln(d)\rfloor-(\ln(s)/\ln(d))-N'_\varepsilon}\}^a/(d^a-1)$$

$$\leq C\{d^{\lfloor \ln(s)/\ln(d)\rfloor-(\ln(s)/\ln(d))-N'_\varepsilon}\}^a$$

$$\leq Cd^{-N'_\varepsilon a}.$$

Thus, if we choose $\lfloor \ln(C/\varepsilon)/(a\ln(d))\rfloor+1$ as the value of N'_ε, then (3A.5) is satisfied for this constant $N'_\varepsilon(\geq 1)$.

In the same manner as the case of (3A.5), we can show that (3A.6) holds for the same constant N'_ε.

Next, we show that (3A.7) is satisfied for a constant $N''_\varepsilon(\geq 1)$. Since $n\geq\lfloor \ln(s)/\ln(d)\rfloor$, it holds that

$$0\leq sd^{-n}=d^{(\ln(s)/\ln(d))}d^{-n}$$

$$=d^{(\ln(s)/\ln(d))-n}$$

$$\leq d^{1+\lfloor \ln(s)/\ln(d)\rfloor-n}$$

$$\leq d^{1-1}$$

$$=1.$$

From (3A.2) of the condition (C-2), we can assume that

$$|g_s(sd^{-n})|\leq C(sd^{-n})^b$$

$$=Cs^bd^{-nb}.$$

Therefore,

$$\sum_{n\geq\lfloor \ln(s)/\ln(d)\rfloor+N_\varepsilon''} |g_s(sd^{-n})| \leq \sum_{n\geq\lfloor \ln(s)/\ln(d)\rfloor+N_\varepsilon''} Cs^b(d^{-b})^n$$

$$=Cs^b\{(d^{-b})^{\lfloor \ln(s)/\ln(d)\rfloor+N_\varepsilon''}\}/(1-d^{-b})$$

$$=C(d^{(\ln(s)/\ln(d))-\lfloor \ln(s)/\ln(d)\rfloor-N_\varepsilon''})^b/(1-d^{-b})$$

$$\leq C(d^{(\ln(s)/\ln(d))-\lfloor \ln(s)/\ln(d)\rfloor-N_\varepsilon''})^b$$

$$\leq C(d^{1-N_\varepsilon''})^b.$$

Thus, if we choose $\lfloor \ln(C/\varepsilon)/(b\ln(d))\rfloor+2$ as the value of N_ε'', then (3A.7) is satisfied for this constant $N_\varepsilon''(\geq 1)$.

Also, we can prove (3A.8) for the same constant N_ε'' in the same manner as the case of (3A.7).

Here, if we define N by

$N_\varepsilon=\max\{\lfloor \ln(C/\varepsilon)/(a\ln(d))\rfloor+1,\ \lfloor \ln(C/\varepsilon)/(b\ln(d))\rfloor+2\}$,

then (3A.5)-(3A.8) are satisfied for a given $\varepsilon>0$.

From (3A.4)-(3A.8) proved above, it follows that for any $\varepsilon>0$, there exists K_ε such that

$$\left|\sum_{n\geq 1} g_s(sd^{-n})-\sum_{n\geq 1} G(sd^{-n})\right|<5\varepsilon.$$

holds for $s\geq K_\varepsilon$. Hence,

$$\lim_{s\to\infty}\left|\sum_{n\geq 1} g_s(sd^{-n})-\sum_{n\geq 1} G(sd^{-n})\right|=0. \tag{3A.9}$$

From the fact that $s=d^{\ln(s)/\ln(d)}$, it holds that

$$\sum_{n\geq 1} G(sd^{-n})=\sum_{n\geq 1} G(d^{(\ln(s)/\ln(d))-n})$$

$$=\sum_{n\geq 1} G(d^{(\ln(s)/\ln(d))-\lfloor \ln(s)/\ln(d)\rfloor+(\lfloor \ln(s)/\ln(d)\rfloor-n)}). \tag{3A.10}$$

From (3A.1) of the condition (C-2), we can show that

$$\lim_{s\to\infty} \Big| \sum_{n\geq 1} G\left(d^{(\ln(s)/\ln(d))-\lfloor \ln(s)/\ln(d)\rfloor+(\lfloor \ln(s)/\ln(d)\rfloor-n)}\right)$$

$$-\sum_{p=-\infty}^{\infty} G\left(d^{(\ln(s)/\ln(d))-\lfloor \ln(s)/\ln(d)\rfloor-p}\right)\Big|=0. \qquad (3A.11)$$

Finally, (3A.9)-(3A.11) lead to the lemma which we set up to prove. □

Appendix 3-B Numerical computaions of $g(\alpha)$

When $x<1$, we expand the exponential functions into power series in $F(x)$ aiming at the numerical stability;

$$F(x)=(m-1)(-e^{-mx}+1)-e^{-x}+1-(e^{-mx}-1+mx)/x$$

$$=-(m-1)\sum_{n\geq 1}\frac{(-mx)^n}{n!}-\sum_{n\geq 1}\frac{(-x)^n}{n!}-\sum_{n\geq 2}\frac{(-mx)^n}{n!}/x$$

$$=\sum_{n\geq 1}(1/n!)[\{-(m-1)+m/(n+1)\}m^n-1](-1)^n x^n.$$

Since $p\geq 1$ gives $(Qm)^{\alpha-p}<1$ for α such that $0\leq\alpha<1$, we have

$$\sum_{p\geq 1}F((Qm)^{\alpha-p})=\sum_{n\geq 1}\frac{(-1)^n}{n!}[\{-(m-1)+m/(n+1)\}m^n-1](Qm)^{(\alpha-1)n}/(1-(Qm)^{-n}). \tag{3B.1}$$

Next, we consider the case that $x\geq 1$. $F(x)$ is rewritten as

$$F(x)=-(m-1+1/x)e^{-mx}-e^{-x}+1/x. \tag{3B.2}$$

When $x\geq 1$, we use $F(x)$ given by Eq.(3B.2) without expanding the exponential functions into power series. The following property may be useful in calculating $F(x)$ with a large value of x: i.e., compared with the last term $(1/x)$ of Eq.(3B.2), the remains promptly go to zero as x becomes large.

Appendix 3-C Proof of Corollary 3.1

Corollary 3.1

$$\lim_{k\to\infty} \Big| A(k,Q,m)/k - \Big[(1/\ln(Qm)) \sum_{p=-\infty}^{\infty} \{(m^{(2\pi p i/\ln(Qm))}-1)/(-2\pi p i/\ln(Qm))$$

$$+(m^{1+(2\pi p i/\ln(Qm))})/(1+(2\pi p i/\ln(Qm))\}$$

$$\cdot\Gamma(1-(2\pi p i/\ln(Qm)))e^{2\pi p i\{(\ln(k)/\ln(Qm))-\lfloor \ln(k)/\ln(Qm)\rfloor\}}\Big]\Big|=0,$$

where $\Gamma(x)$ is the gamma function.

Proof From Poisson's summation formula, for any integrable and smooth function $g(x)$, it is generally satisfied that

$$\sum_{p=-\infty}^{\infty} g(x+p)=\sum_{p=-\infty}^{\infty} \{\underbrace{\int_{-\infty}^{\infty} e^{-2\pi y p i} g(y)dy}_{(A)}\}e^{2\pi x p i}. \tag{3C.1}$$

Now, we define $g(x)$ by $F((Qm)^x)$, and consider the part (A) of Eq. (3C.1).

$$\int_{-\infty}^{\infty} g(y)e^{-2\pi y p i}dy=\int_{-\infty}^{\infty} F((Qm)^y)e^{-2\pi y p i}dy$$

$$=(1/\ln(Qm))\int_{0}^{\infty}(F(x)/x)e^{-(2\pi \ln(x)p/\ln(Qm))i}dx, \tag{3C.2}$$

where we defined x by $(Qm)^y$ for obtaining the final equation of Eq.(3C.2). Further, noting that

$$e^{-(2\pi\ln(x)p)/\ln(Qm)i}=x^{-2\pi p i/\ln(Qm)},$$

we can rewrite Eq.(3C.2) in the form

$$\int_{-\infty}^{\infty} g(y)e^{-2\pi y p i}dy=(1/\ln(Qm)) \int_{0}^{\infty} F(x)x^{\rho-1}dx, \tag{3C.3}$$

where

$$\rho=-(2\pi p/\ln(Qm))i. \tag{3C.4}$$

We will show that the integral in the right-hand side of Eq.(3C.3) is convergent for $|Re(\rho)|<1$. In the first place, we denote this integral by $f(\rho)$; further we divide $f(\rho)$ into the following two parts:

$$f(\rho)=\underbrace{\int_0^1 F(x)x^{\rho-1}dx}_{(B)}+\underbrace{\int_1^{\infty} F(x)x^{\rho-1}dx}_{(C)}. \tag{3C.5}$$

From Lemma 3.2, we have

$$|F(x)|\le C_2/x \qquad \text{for } x\ge 1,$$

$$|F(x)|\le C_3 x \qquad \text{for } 0\le x\le 1.$$

Hence the part (B) yields the estimate

$$(B)=\int_0^1 F(x)x^{\rho-1}dx\le\int_0^1 |F(x)x^{\rho-1}|dx$$

$$\le\int_0^1 (C_3x)x^{Re(\rho)-1}dx=[C_3x^{Re(\rho)+1}/(Re(\rho)+1)]_0^1.$$

Since $Re(\rho)+1>1$, (B) is convergent. Next, in a similar way, we can estimate the part (C) as follows:

$$(C)=\int_1^{\infty} F(x)x^{\rho-1}dx\le\int_1^{\infty}|F(x)x^{\rho-1}|dx$$

$$\le\int_1^{\infty}(C_2/x)x^{Re(\rho)-1}dx=[C_2x^{Re(\rho)-1}/(Re(\rho)-1)]_1^{\infty}.$$

Since $Re(\rho)-1<0$, (C) is also convergent. Therefore, $f(\rho)$ is convergent as well as continuous for $|Re(\rho)|<1$.

First, we will obtain $f(\rho)$ for $0<Re(\rho)<1$. From the definition of $F(x)$, we have

$$f(\rho)=\int_0^{\infty}\{(1-m)e^{-mx}-e^{-x}-(e^{-mx}-1)/x\}x^{\rho-1}dx$$

$$=\underbrace{\int_0^{\infty}(1-m)e^{-mx}x^{\rho-1}dx}_{(D)}-\underbrace{\int_0^{\infty}e^{-x}x^{\rho-1}dx}_{(E)}-\underbrace{\int_0^{\infty}\{(e^{-mx}-1)/x\}x^{\rho-1}dx}_{(F)}. \tag{3C.6}$$

From the definition of the gamma function, the parts (D) and (E) of

Eq.(3C.6) become

$$(D)=(1-m)m^{-\rho}\Gamma(\rho), \tag{3C.7}$$

$$(E)=\Gamma(\rho). \tag{3C.8}$$

On the other hand, by a change of variable, i.e., y=mx the part (F) can be written in the form

$$(F)=m^{1-\rho}\int_0^\infty\{(e^{-y}-1)/y\}y^{\rho-1}dy.$$

Further, we have

$$(F)=m^{1-\rho}\int_0^\infty(-\int_0^1 e^{-\theta y}d\theta)y^{\rho-1}dy=-m^{1-\rho}\int_0^1\{\int_0^\infty e^{-\theta y}y^{\rho-1}dy\}d\theta.$$

$$=-m^{1-\rho}\int_0^1\Gamma(\rho)\theta^{-\rho}d\theta$$

$$=-m^{1-\rho}\Gamma(\rho)/(1-\rho). \tag{3C.9}$$

By substituting Eqs.(3C.7), (3C.8) and (3C.9) in Eq. (3C.6), we finally obtain the following equation for $0<\mathrm{Re}(\rho)<1$

$$f(\rho)=\{(m^{-\rho}-1)/\rho+m^{1-\rho}/(1-\rho)\}\Gamma(\rho+1). \tag{3C.10}$$

Next, we consider $f(\rho)$ for $\mathrm{Re}(\rho)=0$ since given by Eq.(3C.4) satisfies $\mathrm{Re}(\rho)=0$. The right-hand side of Eq.(3C.10) converges when $\mathrm{Re}(\rho)\to 0$ and then $f(\rho)$ is continuous for $|\mathrm{Re}(\rho)|<1$ as stated earlier; therefore it is obvious that $f(\rho)$ is also given by Eq.(3C.10) for $\mathrm{Re}(\rho)=0$.

Consequently, we can obtain

$$\sum_{p=-\infty}^{\infty} F((Qm)^{x+p})=\sum_{p=-\infty}^{\infty} F((Qm)^{x-p})$$

$$=(1/\ln(Qm))\sum_{p=-\infty}^{\infty}\{(m^{(2\pi p i/\ln(Qm))}-1)/(-2\pi p i/\ln(Qm))$$

$$+(m^{1+(2\pi p i/\ln(Qm))})/(1+(2\pi p i/\ln(Qm))\}$$

$$\cdot\Gamma(1-(2\pi p i/\ln(Qm)))e^{2\pi p x i} \tag{3C.11}$$

From Theorem 3.1 and Eq.(3C.11), the corollary is proved by setting the value of x at $(\ln(k)/\ln(Qm))-\lfloor \ln(k)/\ln(Qm)\rfloor$. □

Appendix 3-D Throughput performance of the basic tree algorithm

The maximum throughput of the Q-ary basic tree algorithm has already been analyzed in [MATH 85]. This appendix provides the same results using our approach given in section 3.4; these results will be used in chapter 5.

As with the case of TA/Ms, we define the following quantities in the basic tree algorithm:

$$S_{inf}(Q) = \lim_{k\longrightarrow\infty} \inf \frac{k}{T_k},$$

$$S_{sup}(Q) = \lim_{k\longrightarrow\infty} \sup \frac{k}{T_k}.$$

From Eq.(3.11) and the definition of A(k,Q,m), we have

$$T_k = 1+Q+A(k,Q,1)Q.$$

Thus, using the definitions (3.17) and (3.18) concerning $g(\alpha)$, $S_{inf}(Q)$ and $S_{sup}(Q)$ are give by

$$S_{inf}(Q) = \left(\frac{1}{Q}\right)\frac{1}{\sup_{0\leq\alpha<1} g(\alpha)},$$

$$S_{sup}(Q) = \left(\frac{1}{Q}\right)\frac{1}{\inf_{0\leq\alpha<1} g(\alpha)}.$$

Furthermore, we define S*(Q) corresponding to S*(Q,m) of TA/Ms as follows:

$$S^*(Q) = 1/(Q\cdot\bar{A}(Q,1))$$

$$= \frac{\ln(Q)}{Q},$$

where $\bar{A}(Q,1)$ is given by Eq.(3.21).

Table 3.6 shows $S_{inf}(Q)$, $S^*(Q)$ and $S_{sup}(Q)$ for several values of Q. We see that $S^*(Q)$ is used for the stable maximum throughput in the cases that Q is small; these cases are important because they provide excellent performance in the basic tree algorithm as shown in Table 3.6.

Chapter 4

Performance Evaluation of

Free Access Tree algorithm with Mini-Slots

4.1. Introduction

Among a variety of collision resolution algorithms that have been investigated, a class of tree algorithms [CAPE 79, TSYB 78] is one of the outperforming algorithms. Tree algorithms broadly divide into two classes depending on how new packets are handled: free access [TSYB 80b, FAYO 85, MATH 85] and blocked access [TSYB 78, CAPE 79, MATH 85] tree algorithms.

Free Access (FA) tree algorithms do not distinguish between new and collided packets. Users attempt to transmit a new packet immediately after its generation. Thus, users are not required to monitor the channel continuously; They sense the channel only when they have packets to transmit (i.e, limited sensing [HUMB 86]). Due to the simplicity and ease of implementation, FA tree algorithms are of practical interest. The upper bound of the maximum throughput in the context of free access has recently been shown to be 0.567 [HUMB 86]. This is a tight bound if collided packets are retransmitted in some fashion to avoid a collision with other previously collided packets. However, no specific algorithm has been found yet to achieve the maximum throughput close to this bound.

On the other hand, blocked access (BA) tree algorithms [GALL 78, MASS 80, MOSE 85] force new packets to wait until all the outstanding collisions have been resolved, and thus require users to monitor the channel continuously. Mosely and Humblet [MOSE 85] have shown that the maximum throughput of a BA tree algorithm is 0.48776, assuming that the users distinguish between an empty, a successful and a collided slots. If users can further detect the multiplicity of a collision, i.e., the number of packets involved in a collision, the maximum throughput increases to 0.53237 [GEOR 83, TSYB 80a].

As stated in chapter 3, to improve the performance of a BA tree algorithm, mini-slots were introduced to provide better feedback on the channel status with network users; so called BA tree algorithms with mini-slots (BA Q-ary TA/M). In this class of algorithms, Q number of mini-slots are provided within a (large) slot to allow users to acquire additional information on the state of the packet transmission (see Fig.4.1). Data sub-slot length is equal to packet transmission time. When a user sends a packet (using a data sub-slot in a large slot), he also sends a signal in a mini-slot randomly chosen. In case of a collision, the current enabled set of users (the set of users who currently have transmission right) are divided into Q number of subtrees, each corresponding to a group of users who have chosen the same mini-slot. BA Q-ary TA/Ms have been investigated assuming binary (i.e., something/nothing) [SZPA 85, HUAN 85] and ternary (idle/success/collision) [MERA 83] feedback information in a mini-slot, and they have been shown to provide excellent performance.

In this chapter, we introduce mini-slots into a free access (FA) tree algorithm in order to improve its performance(+). We consider two types of feedback information in a mini-slot; binary (something/nothing) and ternary (idle/success/collision) feedback information. Maximum throughput and the average transmission delay are analyzed for an FA TA/M.

(+) Combining mini-slots with reservation scheme will be treated in chapter 5. In this chapter, we are interested in an FA TA/M coupled with direct channel access scheme (i.e., to send a packet directly in a data sub-slot and to use mini-slots to resolve a collision) because of its simplicity and of its practical importance.

The exact description of the algorithm is presented in section 4.2. Section 4.3 analyzes maximum throughput of the algorithm. The upper bound on the maximum throughput in the whole class of free access algorithms (including tree type algorithms and others) is also obtained as the asymptotic case where Q approaches infinity. In section 4.4, the lower bound on the average transmission delay of an FA TA/M is analytically obtained. This lower bound is also a lower bound on the average transmission delay in the whole class of free access algorithms. In section 4.5, numerical examples are provided as well as the optimal value of Q (the number of mini-slots in a slot) to achieve the highest throughput for a fixed value of a mini-slot length.

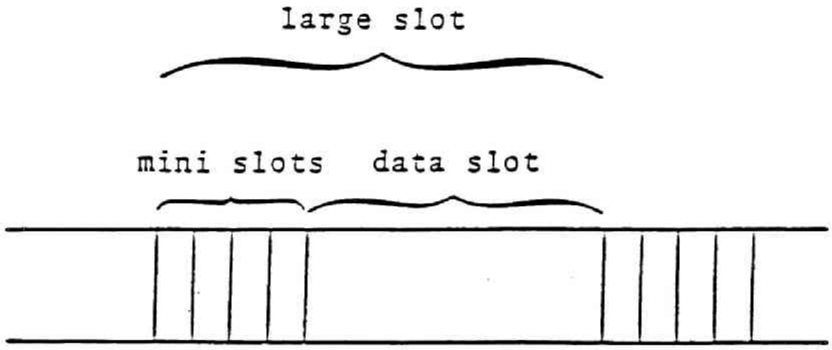


Figure 4.1 Slot configuration

4.2. Free Access Q-ary Tree Algorithms with Mini-Slots

In our analysis, we assume that the time is slotted, and a (large) slot consists of Q number of mini-slots and a data sub-slot (Fig.4.1). The size of a data sub-slot is equal to transmission time of a packet.

We consider a Free Access Q-ary Tree Algorithm with Mini-slots (FA TA/M). In free access algorithms, newly generated packets, regardless of if they arrive while collisions are being resolved, are transmitted immediately after their generation. We study two cases regarding the feedback information available in a mini-slot; ternary and binary feedback information.

FA TA/M with Ternary Feedback (FA TA/M-TF);

Users distinguish between an empty mini-slot (i.e, no user is sending a signal), a successful mini-slot (i.e., only one user is sending a signal) and a collided mini-slot (i.e., more than one user is sending a signal); this is often referred to as 0, 1, e-ternary feedback [BERG 84].

FA TA/M with Binary Feedback (FA TA/M-BF);

Users have limited capability to detect the signal level of a mini-slot so that users can only distinguish between an empty mini-slot (i.e., no signal detected) and a busy mini-slot (i.e., signal detected); namely, something/nothing binary feedback [BERG 84].

In Q-ary TA/M with ternary feedback, a user sends a packet (in a data sub-slot of a large slot), he also sends a signal in a mini-slot randomly chosen. In case of collision, the current enabled set of users are first partitioned into Q number of sub-sets, each corresponding to a group of users who have chosen the same mini-slot. Non-active sub-sets (i.e., sub-sets which correspond to an empty mini-slot) are deleted from the further collision resolution process. Each active sub-set with only one active user (i.e., a sub-set which corresponds to a successful mini-slot) constitutes a new subtree. Each active sub-set with more than one active user (i.e., a sub-set which corresponds to a collided mini-slot) is further divided into m subtrees, where each active user is randomly assigned to one of the m new subtrees. (This algorithm will be referred to as FA TA/M-TF(m) in the following.) Thus, this division process results in $s+m\times c$ number of new subtrees, where s and c are the numbers of successful and collided mini-slots in a large slot, respectively. One subtree is chosen from these $s+m\times c$ new subtrees for collision resolution in the subsequent (large) slot. If further collision occurs, the enabled set is continually divided in the same manner until the collision is resolved. Note that, since we assume a free access algorithm, newly generated packets are immediately transmitted even when the system is in a collision resolution process.

In a Q-ary TA/M with binary feedback, users do not distinguish a successful and a collided mini-slots, and hence, all the active subsets, regardless of how many active users there are in each of them, are treated in the same way; each active subset is randomly divided into m subtrees. (This algorithm will be referred to as FA TA/M-BF(m).) Thus, this division process results in $(s+c)\times m$ number of new subtrees.

Figures 4.2 and 4.3 illustrate a collision resolution process in an FA TA/M-TF(2) and an FA TA/M-BF(1), respectively. In both figures, four users (A, B, C, D) collided in slot 1. Users A and B have chosen the first mini-slot to send signal, and C and D have chosen the third mini-slot, resulting in no successful mini-slots and two collided mini-slots; i.e., s=0 and c=2. In the TA/M-TF(2) (Fig.4.2), the users involved in the initial collision are partitioned into 4 (i.e., $s+c\times m=4$) enabled subtrees. Note that non-active subset (corresponding to the 2nd mini-slot) has been removed from the collision resolution process. Let us assume these subtrees are; a subtree rooted at node 2 with no active users, a subtree rooted at node 3 with users A and B, a subtree at node 6 with user C, and a subtree at node 9 with user D. Since subtree 2 has no active users, the slot (slot 2) assigned to this subtree remains unused. Subtree 3 results in collision again. User C in subtree 6 collides with a new packet at user E. (Note we have assumed free access algorithm.) Subtrees 3 and 6 are again divided for further collision resolution. Subtree 9 results in a successful transmission.

The TA/M-BF(1), upon detecting the initial collision, partitiones the users into two (i.e., $(s+c)\times 1=2$) subtrees a subtree rooted at node 2 (with active users A and B) and a subtree rooted at node 6 (with active users C and D) (see Fig.4.3). Both subtrees 2 and 6 lead to collision again. The same collision resolution process is repeated until all the outstanding collided packets are removed from the system.

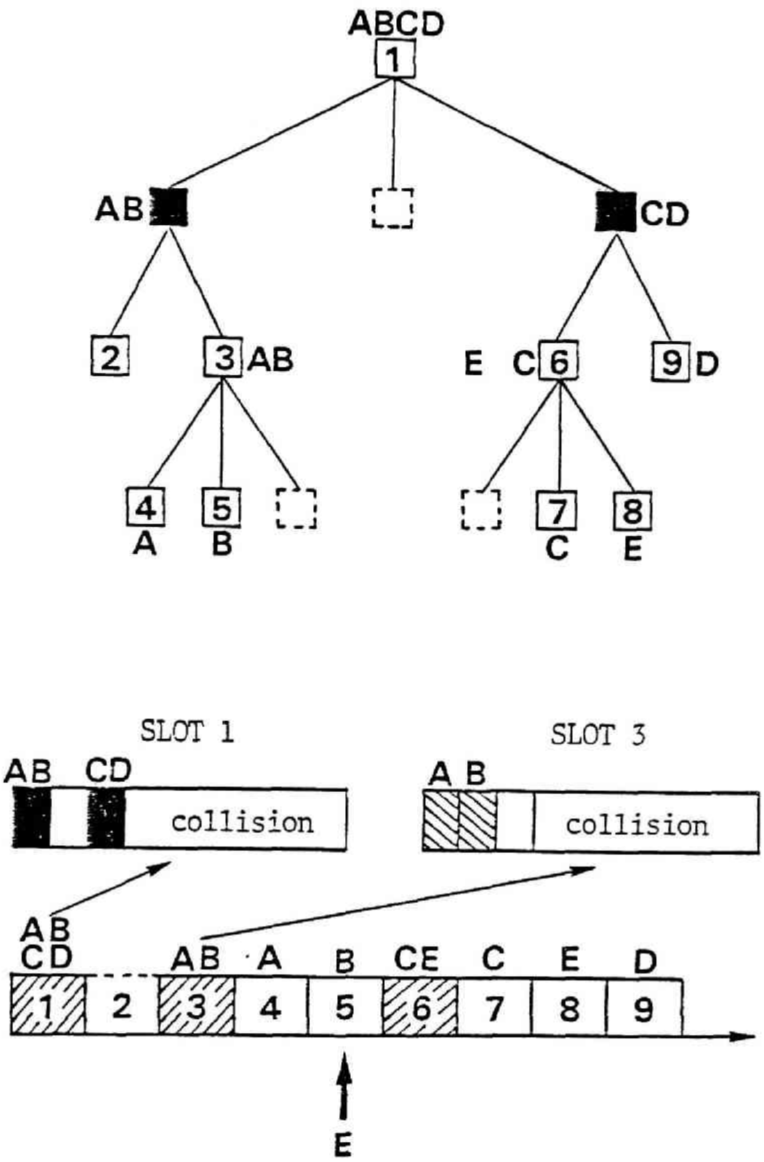


Figure 4.2 Example of transmission process in FA TA/M-TF(2) with Q=3

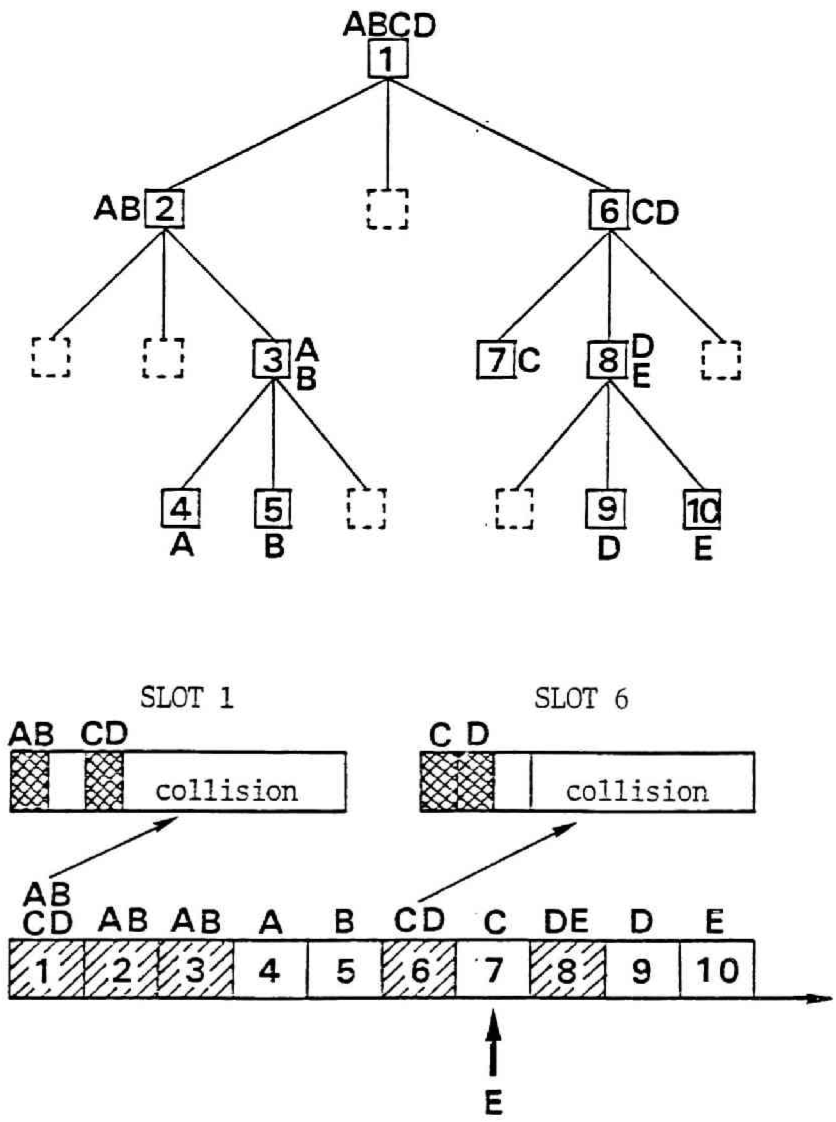


Figure 4.3 Example of transmission process in FA TA/M-BF(1) with Q=3

4.3. Analysis of Throughput Characteristics

4.3.1 Maximum Throughput of Q-ary FA TA/Ms

We analyze Q-ary FA TA/M-BF(m) and Q-ary FA TA/M-TF(m) to obtain its throughput characteristic (in this section) and the average transmission delay (in section 4.4). In this analysis, we assume that unit time is equal to a large slot and that new packet arrivals in different (large) slots are independent and follow an identical Poisson process with rate λ (packets/large slot). We further assume that the propagation delay is zero. Let h denote the ratio of the length of a mini-slot to the data sub-slot length. Since overhead due to mini-slots is hQ, the maximum throughput (per data sub-slot) $\bar{S}(Q,m)$ becomes

$$\bar{S}(Q,m) = \frac{S(Q,m)}{1+hQ},$$

where $S(Q,m)$ is the maximum throughput when there is no mini-slot overhead (i.e., when h is equal to zero). In the following, we will obtain $S(Q,m)$.

We first consider the collision resolution time (CRT), the time required to resolve a collision, given that k number of packets are involved in the collision. Let M_k be a conditional average CRT for a given collision multiplicity k. Namely,

$$M_k = \sum_{i\geq 1} i \text{ Prob}[\text{CRT}=i \mid \text{Collision Multiplicity}=k].$$

If k is zero (i.e., idle slot) or one (i.e., successful slot), only one slot is used, and hence, M_k becomes one. If k is greater than or equal to two, collision arises. In this case, collided packets are divided into Q sub-sets. A CRT to resolve the original collision is the sum of the CRTs of these Q sub-sets. Let n_i be the random variable representing the number of packets in sub-set i. Then,

$$P[n_1=N_1,\ldots,n_i=N_i,\ldots,n_Q=N_Q]=\frac{k!}{N_1!\ldots N_Q!}\left(\frac{1}{Q}\right)^k$$

and $\Sigma_{i=1}^{Q} N_i=k$. We denote the above density function as $U(\frac{1}{Q},k)$.

In both binary feedback (FA TA/M-BF(m)) and ternary feedback (FA TA/M-TF(m)) cases, an empty sub-set i ($n_i=0$) among the Q sub-sets yields no new subtree, and hence, CRT for an empty sub-set is 0.

In the ternary feedback case (FA TA/M-TF(m)), sub-set i containing only one packet ($n_i=1$) results in one subtree. Let X_i^j be the number of new arrivals in the slot immediately preceding to the beginning of the collision resolution process of the j^{th} subtree (resulting from the sub-set i). In the particular case where $n_i=1$, there is only one subtree generated, and hence, j assumes only 1. Note that we have assumed that new arrivals are independent of slots, and hence, X_i^j is independent of i and j. We have further assumed that $X_i^j=X$ follows Poisson distribution, namely,

$$p(h) = \frac{\lambda^h}{h!}e^{-\lambda}.$$

Since new packets, in addition to the packet assigned to the subtree, are immediately transmitted in an FA algorithm, CRT for the original sub-set i is the time required to resolve a collision of multiplicity $1+X_i^1$. Namely, CRT in this case becomes $M_{1+X_i^1}$.

On the other hand, in the binary feedback case (FA TA/M-BF(m)), a sub-set i containing only one packet, say test packet, results in m subtrees. Of these m subtrees, CRT of the one, say the first subtree, to which the test packet has been assigned to becomes $M_{1+X_i^1}$; for each of the remaining subtrees (h^{th} subtree), CRT becomes $M_{X_i^h}$. Hence, CRT for the sub-set i ($n_i=1$) becomes $M_{1+X_i^1}+\Sigma_{h=2}^{m}M$.

In both FA TA/M-TF(m) and FA TA/M-BF(m), a sub-set i containing more than one packet ($n_i\geq 2$) generates m subtrees. In this case, n_i number of packets are randomly spread over the m subtrees. If we let r_i^j be the number of packets assigned to the j^{th} subtree (of these m subtrees), CRT of the j-th subtree becomes $M_{r_i^j+X_i^j}$, where X_i^j is defined above. Hence, CRT for the original sub-set i becomes $\Sigma_{j=1}^{m} M_{r_i^j+X_i^j}$. Note that $\Sigma_{j=1}^{m} r_i^j=n_i$ and that the density function of r_i^j is given by $U(\frac{1}{m},n_i)$.

From the above arguments, M_k becomes

$$M_k \begin{cases} = 1 & (k=0,\ 1) \\ = 1 + \sum\limits_{\{i|n_i=0\}} 0 + \sum\limits_{\{i|n_i=1\}} \{M_{1+X_i^1} + \delta_B \sum\limits_{h=2}^{m} M_{X_i^h}\} & \\ \qquad + \sum\limits_{\{i|n_i\geq 2\}} \sum\limits_{n=1}^{m} M_{r_i^j+X_i^j} & (\text{otherwise}). \end{cases}$$

In the above equation, $\sum\limits_{\{i|A\}}$ represents the sum over i-s satisfying the condition A. δ_B depends on the algorithm and is

$$\delta_B = \begin{cases} 1 & (\text{TA/M-BF(m)}) \\ 0 & (\text{TA/M-TF(m)}). \end{cases}$$

Thus, M_k satisfies the following equation:

$$M_k=1+\sum_{n_1+\ldots+n_Q=k} U(\frac{1}{Q},k)\sum_{i=1}^{Q} v_{n_i}, \qquad (k\geq 2) \tag{4.1}$$

where $\sum\limits_{n_1+\ldots+n_Q=k}$ represents the sum over all possible combinations of $n_1,\ldots,n_Q$ such that $n_1+\ldots+n_Q=k$, and v_{n_i} is given by

$$v_{n_i} \begin{cases} = 0 & (n_i=0) \\ = \sum\limits_{h\geq 0} M_{1+h}p(h) + \delta_B(m-1)\sum\limits_{h\geq 0} M_h p(h) & (n_i=1) \\ = \sum\limits_{r_i^1+\ldots+r_i^m=n_i} U(\frac{1}{m},n_i)\sum\limits_{h\geq 0}\sum\limits_{j=1}^{m} M_{r_i^j+h}p(h) & (n_i\geq 2). \end{cases} \tag{4.2}$$

To obtain Eq.(4.2), we have used that X_i^j is independent of i and j and follows Poisson distribution and that r_i^j has the distribution $U(\frac{1}{m},n_i)$.

Equation (4.1) is rewritten as (see Appendix A of [HUAN 85] for the detail)

$$M_k=1+\sum_{n=0}^{k}\binom{k}{n}(1-\frac{1}{Q})^{k-n}(\frac{1}{Q})^{n-1}v_n. \qquad (k\geq 2)$$

Furthermore, substituting Eq.(4.2) into the above equation, we have

$$M_k= 1+k(1-\frac{1}{Q})^{k-1}[\sum_{j\geq 0}M_{1+j}p(j) + \delta_B k(m-1)\sum_{j\geq 0}M_j p(j)]$$
$$+\sum_{n=2}^{k}\binom{k}{n}(1-\frac{1}{Q})^{k-n}(\frac{1}{Q})^{n-1}\sum_{j\geq 0}p(j)\sum_{i=0}^{n}\binom{n}{i}(1-\frac{1}{m})^{n-i}(\frac{1}{m})^{i-1}M_{i+j} \quad (k\geq 2), \qquad (4.3)$$

$$M_0=M_1=1.$$

After some manipulation (see Appendix 4-A), Eq.(4.3) becomes

$$M_k= 1+\sum_{j\geq 0}\sum_{i=0}^{k}\binom{k}{i}a^{i-1}(1-a)^{k-i}M_{i+j}p(j)$$
$$-\{a^{-1}(1-\frac{1}{Q})^k+(1-\delta_B)k(m-1)(1-\frac{1}{Q})^{k-1}\}\sum_{j\geq 0}M_j p(j) \quad (k\geq 2), \qquad (4.4)$$

where $a=(Qm)^{-1}$.

We now define the following functions:

$$M(z)=\sum_{k\geq 0}M_k\frac{z^k}{k!}, \qquad (4.5)$$

$$M^*(z)=e^{-z}M(z). \qquad (4.6)$$

Taking the derivative of these functions, we have

$$M^{(1)}(z)=\sum_{k\geq 0}M_{k+1}\frac{z^k}{k!}, \qquad (4.7)$$

$$M^{*(1)}(z)=e^{-z}(M^{(1)}(z)-M(z)).$$

Note that $M^*(\lambda)=\sum_{k\geq 0}M_k\frac{\lambda^k}{k!}e^{-\lambda}$ represents the average CRT under the assumption of a Poisson arrival process. Since this quantity plays a key role in the analysis, we will obtain $M^*(\lambda)$ by applying the method proposed by Mathys et al. [MATH 85].

By multiplying both sides of Eq.(4.4) by $z^k/k!$ and then taking sum of both sides over $k\geq 0$ (see Appendix 4-B), we have

$$M^*(z)-a^{-1}M^*(\lambda+az)=1+a^{-1}M^*(\lambda)f(z)+M^{*(1)}(\lambda)g(z) \qquad (4.8)$$

where

$$f(z)=\begin{cases} -\frac{z}{Q}e^{-z}-e^{-z/Q} & \text{(TA/M-BF(m))} \\ -aze^{-z}-\{1+(\frac{1}{Q}-a)z\}e^{-z/Q} & \text{(TA/M-TF(m))}, \end{cases}$$

$$g(z)=-ze^{-z}.$$

We will solve Eq.(4.8) for $M^*(z)$. First, by differentiating Eq.(4.8) twice with respect to z, we have

$$M^{*(2)}(z)-aM^{*(2)}(\lambda+az)=a^{-1}M^*(\lambda)f^{(2)}(z)+M^{*(1)}(\lambda)g^{(2)}(z).$$

This equation has the following solution [MATH 85]:

$$M^{*(2)}(z)=a^{-1}M^*(\lambda)\sum_{i\geq 0} a^i f^{(2)}(\sigma_M^{[i]}(z))+M^{*(1)}(\lambda)\sum_{i\geq 0} a^i g^{(2)}(\sigma_M^{[i]}(z)), \tag{4.9}$$

where

$$\sigma_M^{[i]}(z)=\lambda\frac{1-a^i}{1-a}+a^i z.$$

By integrating Eq.(4.9), we have (see Appendix 4-C for the derivation of the following equation)

$$M^{*(1)}(z)=a^{-1}M^*(\lambda)\ \Theta^{(1)}(f(.);z)+M^{*(1)}(\lambda)\ \Theta^{(1)}(g(.);z), \tag{4.10}$$

where

$$\Theta^{(1)}(\psi(.);z)=\int_0^z \sum_{i\geq 0} a^{-i}\psi^{(2)}(\sigma_M^{[i]}(z))dz,$$

$$\Theta(\psi(.);z)=\int_0^z \Theta^{(1)}(\psi(.);z)dz.$$

By letting $z=\lambda$ in Eq.(4.10), we find that $M^{*(1)}(\lambda)$ is expressed in terms of $M^*(\lambda)$ as follows:

$$\begin{aligned} M^{*(1)}(\lambda)&=a^{-1}M^*(\lambda)\Theta^{(1)}(f(.);\lambda)/\{1-\Theta^{(1)}(g(.);\lambda)\} \\ &=a^{-1}M^*(\lambda)\ w, \end{aligned} \tag{4.11}$$

where

$$w=\frac{\Theta^{(1)}(f(.);\lambda)}{1-\Theta^{(1)}(g(.);\lambda)}$$

$$=\begin{cases} \frac{e^{(1-1/Q)\mu}}{(1-\mu)Q}-\frac{1}{Q} & \text{(TA/M-BF(m))} \\ \frac{1}{1-\mu}\{a+(\frac{1}{Q}-a)\frac{\mu}{Q}\}e^{(1-1/Q)\mu}-a & \text{(TA/M-TF(m))}, \end{cases}$$

$$\mu = \frac{\lambda}{1-a}.$$

Finally, by substituting Eq.(4.11) into Eq.(4.10) and further integrating the resulting equation, we obtain the following solution to Eq.(4.8):

$$M^*(z)=1+a^{-1}M^*(\lambda)b^*(z), \tag{4.12}$$

where

$$b^*(z)= \Theta(f(.);z)+w\ \Theta(g(.);z).$$

We used $\int_0^z M^{*(1)}(z)dz=M^*(z)-M^*(0)=M^*(z)-1$ to obtain Eq.(4.12).

We now proceed to obtain the throughput of FA TA/Ms. Setting $z=\lambda$ in Eq.(4.12) and solving the resulting equation for $M^*(\lambda)$ yields

$$M^*(\lambda)=1/(1-a^{-1}b^*(\lambda)) \tag{4.13}$$

where

$$b^*(\lambda)\begin{cases} = \sum_{i\geq 0} a^{-i}[-e^{-x/Q}+e^{-y/Q}-a^i\frac{\lambda}{Q}e^{-y/Q}] \\ \quad + (w-\frac{1}{Q}) \sum_{i\geq 0} a^{-i}\{-xe^{-x}+ye^{-y}-a^i\lambda(y-1)e^{-y}\} \quad \text{(TA/M-BF(m))}, \\ = \sum_{i\geq 0} a^{-i}[-\{1+(\frac{1}{Q}-a)x\}e^{-x/Q}+\{1+(\frac{1}{Q}-a)y\}e^{-y/Q} \\ \quad -a^i\lambda\{a+(\frac{1}{Q}-a)\frac{y}{Q}\}e^{-y/Q}] \\ \quad + (w-a) \sum_{i\geq 0} a^{-i}\{-xe^{-x}+ye^{-y}-a^i\lambda(y-1)e^{-y}\}, \quad \text{(TA/M-TF(m))} \end{cases}$$

$$x=\mu(1-a^{i+1}),$$

$$y=\mu(1-a^i).$$

For numerical computations, we give the following by expanding the exponential functions into power series and taking the sum over i in the above equation:

$$b^*(\lambda)\begin{cases} =e^{-\mu/Q} \sum_{j\geq 1} \frac{(\mu/Q)^j}{j!} \frac{1}{Q(j+1)} (a-(1-a)j/(1-a^j))\mu \\ \quad +(w-\frac{1}{Q})e^{-\mu} \sum_{j\geq 1} \frac{\mu^{j+1}}{j!}(\frac{\mu}{j+1}-1)(a-(1-a)j/(1-a^j)). \quad \text{(TA/M-BF(m))} \\ =e^{-\mu/Q} \sum_{j\geq 1} \frac{(\mu/Q)^j}{j!}[\frac{1+(1/Q-a)}{Q(j+1)}-(1/Q-a)](a-(1-a)j/(1-a^j))\mu \\ \quad +(w-a)e^{-\mu} \sum_{j\geq 1} \frac{\mu^{j+1}}{j!}(\frac{\mu}{j+1}-1)(a-(1-a)j/(1-a^j)). \quad \text{(TA/M-TF(m))} \end{cases} \tag{4.14}$$

As explained previously, $M^*(\lambda)$ is the unconditional average CRT assuming that new packets arrive according to a Poisson process with rate λ. Thus, the FA TA/Ms will be stable if the right hand side in Eq.(4.13) is positive and finite. Namely, the system is stable if an input rate λ satisfies the following:

$$a^{-1}b^*(\lambda)<1.$$

Thus, λ which satisfies $a^{-1}b^*(\lambda)=1$ gives the upper bound of the stable input rate. We denote this value of λ by $S(Q,m)$, i.e.,

$$a^{-1}b^*(S(Q,m))=1. \tag{4.15}$$

$S(Q,m)$ is the supremum of throughput. Various values of $S(Q,m)$ are given in Tables 4.1 and 4.2. $S(Q,m)$ increases as Q becomes larger in TA/M-BF(m) (m=1, 2) and TA/M-TF(m) ($m\geq 1$). In contrast, $S(Q,m)$ decreases as Q becomes larger in TA/M-BF(m) ($m\geq 3$).

4.3.2 Upper Bound of the throughput in FA TA/Ms

Now, we will turn our attention to the limiting case where Q is infinity and h is equal to zero. Let S^* denote the throughput in this limiting case; thus, $S^*=\lim_{Q\to\infty} S(Q,m)$. By setting $\lambda=S(Q,m)$ in Eq.(4.14), multiplying the resulting equation by a^{-1} and further letting Q approach infinity, we have

$$\lim_{Q\to\infty} a^{-1}b^*(S(Q,m)) = \begin{cases} m/(1-S^*)\sum_{j\geq 1} \frac{S^{*j+1}}{j!}\left(1-\frac{S^*}{1+j}\right)j. & \text{(TA/M-BF(m))} \\ 1/(1-S^*)\sum_{j\geq 1} \frac{S^{*j+1}}{j!}\left(1-\frac{S^*}{1+j}\right)j. & \text{(TA/M-TF(m))} \end{cases}$$

Since, from Eq.(4.15), the above equation is equal to one, we obtain the following simple expression for S^*:

$$\begin{cases} mS^*e^{S^*}-(m-1)S^*-1=0 & \text{(TA/M-BF(m))} \quad (4.16a) \\ S^*e^{S^*}=1. & \text{(TA/M-TF(m)).} \quad (4.16b) \end{cases}$$

Tables 4.1 and 4.2 also shows values of S^* obtained through Eq.(4.16). We see that TA/M-BF(1) and TA/M-TF(m) ($m \geq 1$) provide the highest throughput 0.56714 in the limiting case. Note that TA/M-BF(1) and TA/M-TF(m) will be stable if an input rate λ satisfies the following inequality:

$$\lambda < e^{-\lambda}. \tag{4.17}$$

It should be noted that 0.56714 (S^* of TA/M-BF(1) and TA/M-TF(m)) is the same value as the achievable maximum throughput obtained by Humblet [HUMB 86] in free access algorithms. We will show that this limiting case in TA/M-TF(m) (including TA/M-BF(1)) provides an optimum algorithm in whole free access algorithms; i.e., the highest throughput and the smallest transmission delay are achieved in the context of free and direct channel access. The rest of this section will be restricted to the FA TA/M-TF(m).

Let Q go to infinity in Eq.(4.4) ($\delta_B=0$). Then, Eq.(4.4) reduces to a much simpler form (see Appendix 4-D)

$$M_k = 1 + k \sum_{j \geq 0} M_{1+j} p(j), \qquad (k \geq 2) \tag{4.18}$$

$$M_0 = M_1 = 1.$$

Equation (4.18) indicates that one slot is used because of an initial collision due to k packets (corresponding to the first term (i.e., 1)) and that those k collided packets are exactly divided into k subtrees for resolution of the collision. In each subtree, a collision due to one previously collided packet and j new packets (i.e., 1+j packets) is further resolved if j is larger than or equal to two; otherwise, no collision arises and a collision resolution procedure terminates in the subtree.

In other words, in this limiting case, each of collided packets is isolated in a subtree; no two users will choose the same mini-slot, and thus there are no collided mini-slots. Namely, collided packets are optimally scheduled in the sense that it will take only k slots to transmit k collided packets successfully if no new packets interfere with their retransmissions. Under the assumption that new packets are transmitted immediately after their arrivals (free access mechanism) and that retransmissions of collided packets are scheduled independently of arrivals of new packets, no other algorithms can give better performance than this limiting case. Namely, the limiting case provides optimum performance (i.e., the highest throughput and the smallest transmission delay) under the above assumption. For this reason, S^* is the highest throughput in the whole class of free access algorithms including both algorithms with and without mini-slots.

We obtain M_k (the conditional average CRT) through Eq.(4.18) as (see Appendix 4-E)

$$M_k=1 + k/(e^{-\lambda} - \lambda) \qquad (k\geq 2), \tag{4.19}$$

$$M_0=M_1=1.$$

Let S_k be the conditional throughput (per slot) over a collision resolution interval (CRI) whose initial collision involves k packets. The average number of packets transmitted during the CRI is readily given by $k+\lambda(M_k-1)$. Thus, S_k is written in terms of M_k as follows

$$\begin{aligned} S_k &= \{k+ \lambda(M_k-1)\}/M_k \\ &= \lambda + (k- \lambda)/M_k. \end{aligned}$$

Note that S_k is an increasing function of k and approaches $e^{-\lambda}$ in the limit as $k \longrightarrow \infty$. In the sequel, inequality (4.17) indicates that the system will be stable if an input rate λ is less than the supremum of throughput ($e^{-\lambda}$), which would be achieved if an infinite number of packets were transmitted at the same time ($k=\infty$). Intuitive explanation of the maximum throughput 0.56714 will be also found in [HUMB 86].

Table 4.1

S(Q,m) in FA Q-ary TA/M-BF(m)

Q	S(Q,1)	S(Q,2)	S(Q,3)	S(Q,4)
2	0.37682	0.42912	0.40600	0.37999
3	0.45555	0.44241	0.40517	0.37402
4	0.48848	0.44787	0.40455	0.37118
5	0.50646	0.45084	0.40414	0.36953
6	0.51778	0.45269	0.40385	0.36845
7	0.52554	0.45396	0.40364	0.36769
8	0.53120	0.45489	0.40348	0.36713
9	0.53551	0.45559	0.40335	0.36669
10	0.53890	0.45614	0.40325	0.36635
11	0.54163	0.45659	0.40317	0.36607
12	0.54388	0.45695	0.40309	0.36583
13	0.54576	0.45726	0.40303	0.36563
14	0.54737	0.45752	0.40298	0.36547
15	0.54875	0.45775	0.40294	0.36532
16	0.54995	0.45794	0.40290	0.36519
17	0.55100	0.45811	0.40286	0.36508
18	0.55193	0.45826	0.40283	0.36498
19	0.55276	0.45840	0.40280	0.36489
20	0.55351	0.45852	0.40278	0.36481
∞	0.56714	0.46073	0.40229	0.36332

Table 4.2

S(Q,m) in FA Q-ary TA/M-TF(m)

Q	S(Q,1)	S(Q,2)	S(Q,3)	S(Q,4)
2	0.37682	0.47105	0.47258	0.46055
3	0.45555	0.50377	0.49996	0.48798
4	0.48848	0.51978	0.51488	0.50394
5	0.50646	0.52931	0.52432	0.51446
6	0.51778	0.53564	0.53085	0.52195
7	0.52554	0.54015	0.53564	0.52755
8	0.53120	0.54353	0.53932	0.53191
9	0.53551	0.54616	0.54222	0.53540
10	0.53890	0.54826	0.54457	0.53826
11	0.54163	0.54998	0.54652	0.54065
12	0.54388	0.55141	0.54815	0.54267
13	0.54576	0.55262	0.54955	0.54440
14	0.54737	0.55366	0.55075	0.54590
15	0.54875	0.55456	0.55180	0.54722
16	0.54995	0.55535	0.55273	0.54838
17	0.55100	0.55604	0.55354	0.54941
18	0.55193	0.55666	0.55427	0.55034
19	0.55276	0.55721	0.55493	0.55117
20	0.55351	0.55771	0.55552	0.55193
∞	0.56714			

4.4. Evaluation of Average Transmission Delay

In this section, we obtain the lower bound of the average transmission delay E[D] of the free access TA/M (see references [MERA 84, FAYO 85] for the analysis of the average transmission delay of FA tree algorithm without mini-slots (so called a stack algorithm)). We define the transmission delay as the time interval beginning at packet generation and ending with completion of its successful transmission. As explained in the previous section, the free access TA/M gives the lower bound of transmission delay when the number of mini-slots (in a large slot) approaches infinity and the length of mini-slot goes to zero in TA/M-BF(1) and TA/M-TF(m). Note that TA/M-BF(1) is equivalent to TA/M-TF(1). Hence, we focus on the delay performance of the free access TA/M-TF(m) in this limiting case.

The transmission delay of a packet divides into two elements; D_1, the time from packet generation to the beginning of its initial transmission, and D_2, the time from the initial transmission to the end of its successful transmission. It is easily seen that the mean $E[D_1]$ of the first element D_1 is given by 1/2, and thus we have

$$E[D] = 1/2 + E[D_2]. \tag{4.20}$$

In the following, we will obtain the average of D_2, $E[D_2]$.

Let a_k and d_k denote the expected number of packets arriving in a collision resolution interval (CRI) with an original collision of multiplicity k and the sum of the transmission delays of these a_k packets, respectively, given that the CRI starts with a collision of multiplicity k. Let us define the following functions:

$$D(z)= \sum_{k\geq 1} d_k \frac{z^k}{k!}$$

$$D^*(z)=e^{-z}D(z)$$

$$A^*(z)=e^{-z}\sum_{k\geq 1} a_k \frac{z^k}{k!}.$$

Since, the time points when a CRI starts form a renewal process in the free access TA/M, $E[D_2]$ is given by the following equation (see [FAYO 85]):

$$E[D_2] = \frac{D^*(\lambda)}{A^*(\lambda)}. \tag{4.21}$$

From its definition, the denominator of Eq.(4.21), $A^*(\lambda)=e^{-\lambda}\sum_{k\geq 1} a_k \frac{\lambda^k}{k!}$ is the average number of arrivals in a CRI and becomes $A^*(\lambda)=\lambda M^*(\lambda)$; $M^*(\lambda)$ is the average CRT and is given by (see Eq.(4E.7) in Appendix 4-E)

$$M^*(\lambda)=(1-\lambda)/(1-\lambda e^{\lambda}),$$

we have

$$A^*(\lambda) = \lambda(1-\lambda)/(1-\lambda e^{\lambda}). \tag{4.22}$$

From the definitions, the numerator $D^*(\lambda)$ of Eq.(4.21) is the accumulated transmission delay experienced by all the packets arriving in a CRI. In order to evaluate $D^*(\lambda)$, we first obtain the recurrence equation for d_k, the sum of the accumulated transmission delays in k number of subtrees. In the limiting case, colliding k packets will be divided into k subtrees, each with only one active user (see Eq.(4.18)). The accumulated transmission delay in a subtree contains three elements: the delay due to an initial collision, the time interval from the initial collision to the beginning of a collision resolution process of the subtree, and the accumulated transmission delay of (1+j) packets, where j denotes the number of new packets joining the subtree. Since all k packets are delayed for one slot owing to an initial collision, the first element is k slots. The second element of a subtree, say i^{th} subtree, is due to the sum of the preceding (i-1) CRTs. The third element is given by d_{1+j}. Therefore, d_k satisfies

$$d_k=k+\sum_{i=1}^{k}\sum_{j\geq 0}(i-1)M_{1+j}p(j)+k\sum_{j\geq 0}d_{1+j}p(j) \quad (k\geq 2) \tag{4.23}$$

$d_0=0, \quad d_1=1.$

By multiplying both sides of Eq.(4.23) by $e^{-z}z^k/k!$ and summing over $k\geq 0$, we have

$$D^*(z)=e^{-z}\sum_{k\geq 1}k\frac{z^k}{k!}+e^{-z}\sum_{k\geq 2}\frac{z^k}{(k-1)!}[D^{(1)}(\lambda)+\frac{k-1}{2}M^{(1)}(\lambda)]e^{-\lambda}$$

$$=z+z(1-e^{-z})D^{(1)}(\lambda)e^{-\lambda}+z^2M^{(1)}(\lambda)e^{-\lambda}/2. \tag{4.24}$$

Since $D^{*(1)}(\lambda)=e^{-\lambda}D^{(1)}(\lambda)-D^*(\lambda)$, Eq.(4.24) becomes

$$D^*(z)=z+z(1-e^{-z})\{D^{*(1)}(\lambda)+D^*(\lambda)\}+z^2M^{(1)}(\lambda)e^{-\lambda}/2. \tag{4.25}$$

Taking the derivative of Eq.(4.25), we obtain the following equation

$$D^{*(1)}(z)=1+(1+(z-1)e^{-z})(D^{*(1)}(\lambda)+D^*(\lambda))+ze^{-\lambda}M^{(1)}(\lambda). \tag{4.26}$$

By letting $z=\lambda$ in Eq.(4.26), we have

$$D^{*(1)}(\lambda)=[e^{\lambda}+\{e^{\lambda}+\lambda-1\}D^*(\lambda)+\lambda M^{(1)}(\lambda)]/(1-\lambda). \tag{4.27}$$

Substituting Eq.(4.27) into Eq.(4.25) and letting $z=\lambda$ in the resulting equation, we obtain

$$D^*(\lambda)=[\lambda(e^{\lambda}-\lambda)+\{2-(1+\lambda)e^{-\lambda}\}\lambda^2M^{(1)}(\lambda)/2]/(1-\lambda e^{\lambda}). \tag{4.28}$$

From Eqs.(4.7) and (4.19) (or see Eqs.(4E.5) and (4E.7) in Appendix 4-E), we have

$$M^{(1)}(\lambda)=e^{2\lambda}/(1-\lambda e^{\lambda}).$$

Substitution of $M^{(1)}(\lambda)$ into Eq.(4.28) yields

$$D^*(\lambda)=[\lambda(e^{\lambda}-1)+\{2-(1+\lambda)e^{-\lambda}\}\lambda^2e^{2\lambda}/(1-\lambda e^{\lambda})/2]/(1-\lambda e^{\lambda}). \tag{4.29}$$

Consequently, from Eqs.(4.20), (4.21), (4.22) and (4.29), we have

$$E[D]=\{2-(1+\lambda)e^{-\lambda}\}\lambda e^{2\lambda}/\{2(1-\lambda)(1-\lambda e^{\lambda})\}$$
$$+(e^{\lambda}-\lambda)/(1-\lambda)+0.5. \tag{4.30}$$

Note that E[D] becomes 1.5 as $\lambda\longrightarrow 0$ and becomes infinity as $\lambda\longrightarrow 0.56714$, which is the supremum of throughput.

4.5. Numerical Results

As stated in section 4.3.1, Tables 4.1 and 4.2 show that $S(Q,m)$ is an increasing function of Q in TA/M-BF(m) (m=1, 2) and TA/M-TF(m), and that $S(Q,m)$ is a decreasing function of Q in TA/M-BF(m) ($m\geq 3$). Among TA/M-BF(m)s, the TA/M-BF(1) is the best algorithm, and, among TA/M-TF(m)s, the TA/M-TF(2) is the best one. The TA/M-TF(2) achieves the highest throughput in the whole range of Q except for Q=2. Figure 4.4 shows $S(Q,m)$ of the TA/M-TF(1) and TA/M-TF(2) as a function of the number of mini-slots along with its upper bound 0.56714. It is seen that the maximum throughput of TA/M-TF(1) (or equivalently TA/M-BF(1)) and TA/M-TF(2) approaches the upper bound as Q increases. In these tables and Fig. 4.4, the length of mini-slot (h) is assumed to be zero.

Tables 4.3 and 4.4 discuss the effect of mini-slot overhead on the throughput performance. As described in section 4.3, the maximum throughput (per data slot) is $\bar{S}(Q,m)=S(Q,m)/(1+hQ)$. Table 4.3 shows $\bar{S}(Q,m)$ as a function of Q for h=0.001 in the FA TA/M-TF(m). Both $S(Q,m)$ and $1+hQ$ are increasing functions of Q (see Fig.4.4 for $S(Q,m)$). Thus, there is an optimum value of Q (Q_{opt}) which maximizes $\bar{S}(Q,m)$ for a given value of h; for instance, Q_{opt} is 23 in TA/M-BF(1) and 19 in TA/M-TF(2) for h=0.001. Table 4.4 shows Q_{opt} and the corresponding $\bar{S}(Q_{opt},m)$ for various values of h. From this table, we see that TA/M achieves throughput very close to the upper bound 0.56714 when h is reasonably small; for instance, TA/M-TF(2) achieves throughput 0.54682 when h=0.001. Noting that 2000 bit data packets and 2 bit long mini-slot gives h=0.001, TA/M achieves throughput close to the upper bound in practical systems using free and direct channel access.

Figures 4.5 and 4.6 show simulation results for the average transmission delay of TA/M-TF(2) and TA/M-BF(1), respectively. Both figures also plot the theoretical lower bound of the average transmission delay given by Eq.(4.30). h is assumed to be zero in both figures. The average transmission delay in both figures approaches the theoretical lower bound as Q increases. In the case that h=0.001, Fig. 4.7 illustrates the optimum average transmission delay of TA/M-TF(2) and TA/M-BF(1); i.e., TA/M-TF(2) with Q=19 and TA/M-BF(1) with Q=23 (see Table 4.2 for an optimum value of Q for h=0.001). This figure shows that if h is reasonably small (i.e., h=0.001), both of TA/M-TF(2) and TA/M-BF(1) provide the average transmission delay close to the lower bound.

As seen in Figs. 4.4, 4.5, 4.6 and 4.7, there is not significant difference in both throughput and delay characteristics between TA/M-BF(1) and TA/M-TF(2). Since that TA/M-BF(1) only requires something-nothing binary feedback in mini-slot and that TA/M-BF(1) is less complex to implement than TA/M-TF(2), we may conclude that TA/M-BF(1) is the more practical from an implementation view point.

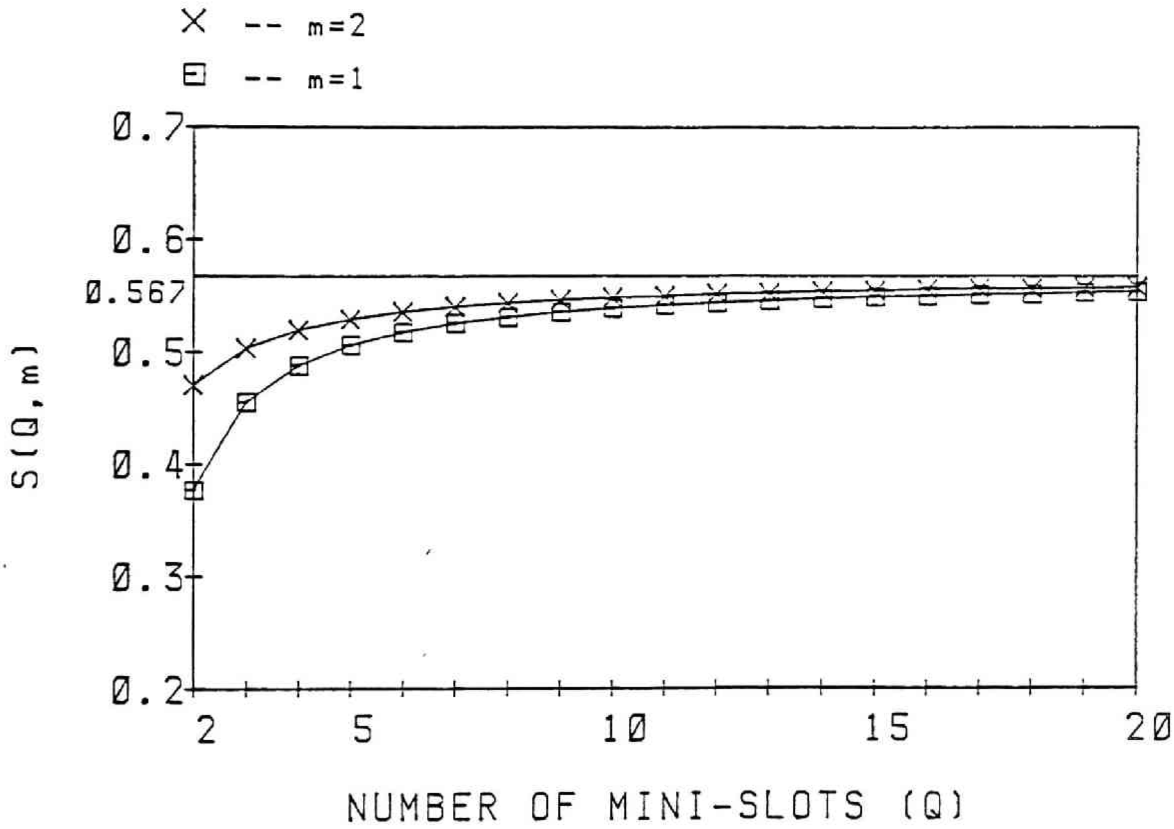


Figure 4.4 S(Q,m) and its upper bound in FA TA/M-TF(m)

Table 4.3

$\bar{S}(Q,m)$ in FA TA/M-TF(m) (h=0.001)

Q	m=1	m=2
2	0.37607	0.47011
3	0.45419	0.50226
4	0.48653	0.51771
5	0.50394	0.52668
6	0.51469	0.53245
7	0.52189	0.53640
8	0.52698	0.53922
9	0.53073	0.54129
10	0.53356	0.54283
11	0.53574	0.54400
12	0.53743	0.54487
13	0.53876	0.54553
14	0.53981	0.54602
15	0.54064	0.54637
16	0.54129	0.54660
17	0.54179	0.54675
18	0.54217	0.54682
19	0.54245	<u>0.54682</u>
20	0.54266	0.54678
21	0.54278	0.54668
22	0.54284	0.54654
23	<u>0.54285</u>	0.54637
24	0.54281	0.54617
25	0.54274	0.54594
26	0.54263	0.54569
27	0.54248	0.54542
28	0.54232	0.54514
29	0.54212	0.54483
30	0.54190	0.54452

Table 4.4

Optimum values of Q and $\bar{S}(Q_{opt},m)$ in FA TA/M-TF(m)

h	Q_{opt}	$\bar{S}(Q_{opt},1)$	Q_{opt}	$\bar{S}(Q_{opt},2)$
0.001	23	0.54285	19	0.54682
0.002	16	0.53290	13	0.53862
0.003	14	0.52531	11	0.53241
0.004	12	0.51897	9	0.52718
0.005	11	0.51339	9	0.52264
0.006	10	0.50840	8	0.51864
0.007	9	0.50377	7	0.51492
0.008	9	0.49954	7	0.51151
0.009	8	0.49552	6	0.50820
0.010	8	0.49185	6	0.50532
0.011	8	0.48824	6	0.50248
0.012	7	0.48482	6	0.49966
0.013	7	0.48170	5	0.49700
0.014	7	0.47863	5	0.49468
0.015	7	0.47560	5	0.49238
0.016	7	0.47261	5	0.49010
0.017	6	0.46985	5	0.48784
0.018	6	0.46731	5	0.48561
0.019	6	0.46479	5	0.48339
0.020	6	0.46230	4	0.48128

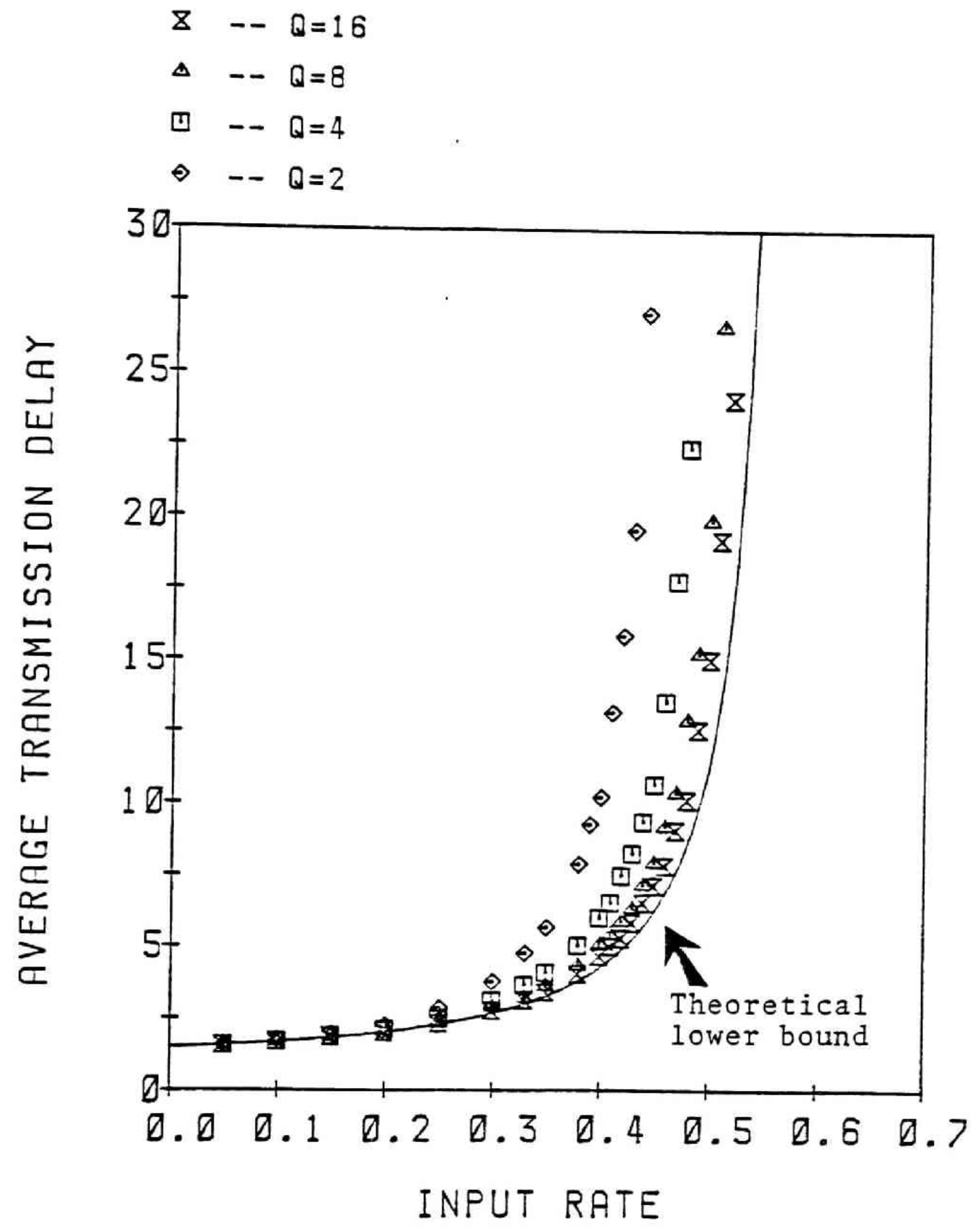


Figure 4.5 Average transmission delay of FA TA/M-TF(2) (h=0)

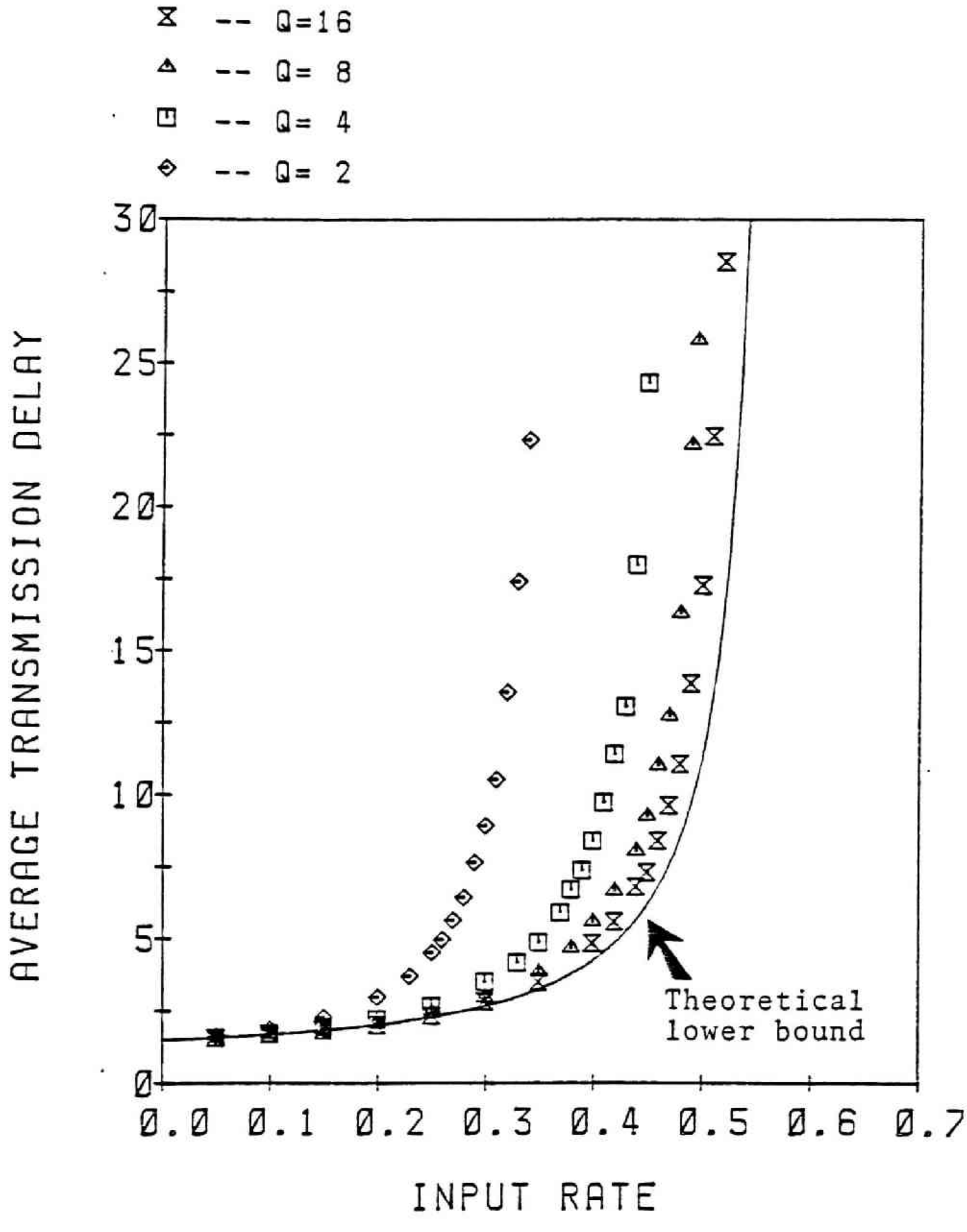


Figure 4.6 Average transmission delay of FA TA/M-BF(1) (h=0)

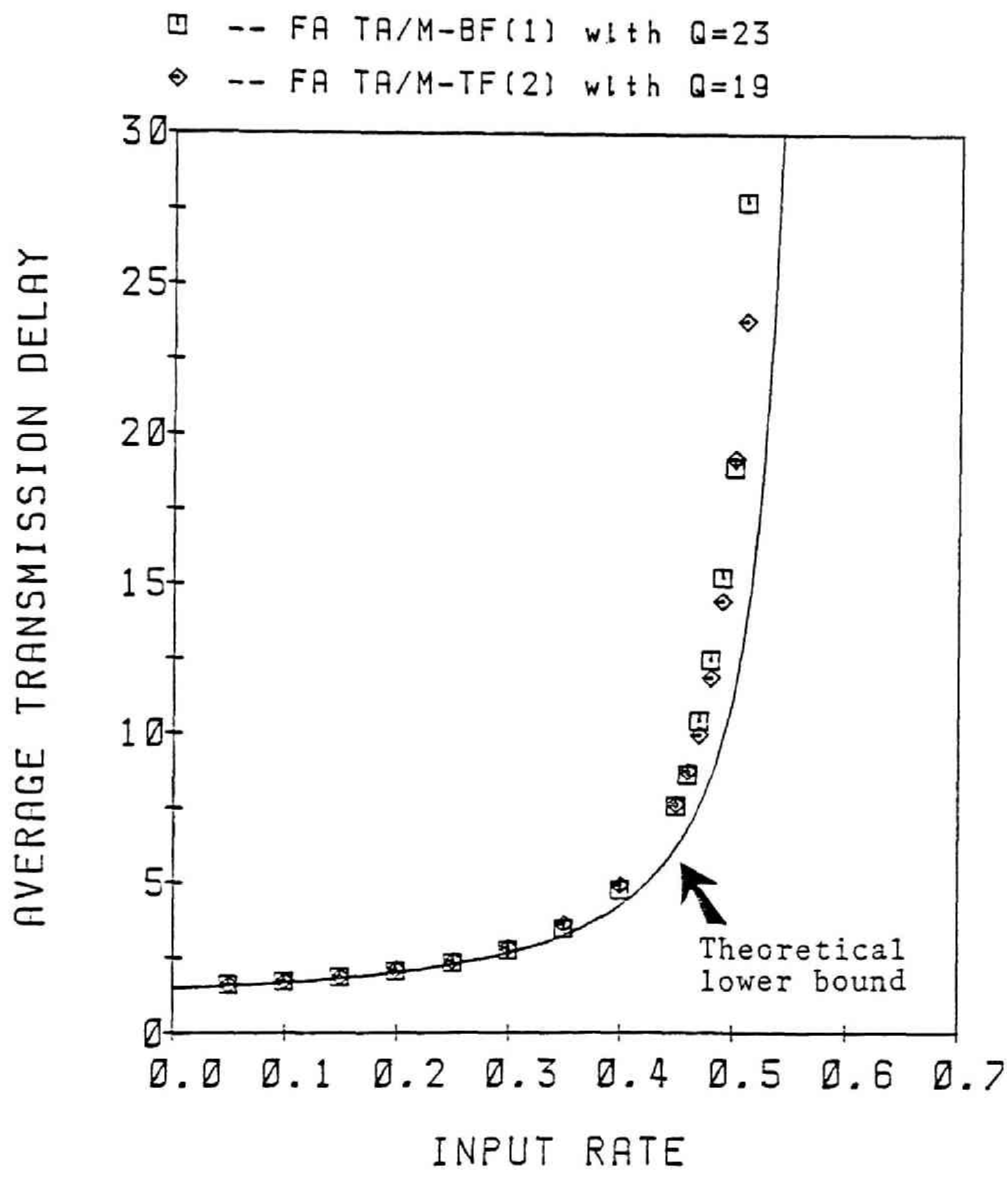


Figure 4.7 Average transmission delay of FA TA/M-BF(1) and FA TA/M-TF(2) (h=0.001)

4.6. Conclusions

In this chapter, we have studied a free access tree algorithm with mini-slots. In the analysis, we have assumed that the binary or the ternary feedback information is available in a mini- slot. For both TA/M-BF and TA/M-TF, our analysis provides the following three performance measures:

(1) the maximum throughput,

(2) the upper bound of throughput, and

(3) the lower bound of the average transmission delay.

We also presented simulation results of the average transmission delay.

As explained in section 4.3, the lower bound of the average transmission delay obtained in this paper is also the lower bound of the delay in the whole class of free access algorithms including both algorithms with and without mini-slots. Since there is not significant difference in performance between TA/M-BF and TA/M-TF, both algorithms give a practical way to achieve performance close to the best performance of all the free access algorithms.

Appendix 4-A Derivation of Eq.(4.4)

Eq.(4.3) can be rewritten as

$$M_k = 1+k(1-\frac{1}{Q})^{k-1}\sum_{j\geq 0} M_{1+j}p(j)+\delta_B k(1-\frac{1}{Q})^{k-1}(m-1)\sum_{j\geq 0} M_j p(j)$$
$$+Q\sum_{n=2}^{k}\binom{k}{n}(1-\frac{1}{Q})^{k-n}(\frac{1}{Q})^n\sum_{j\geq 0}p(j)\{m\sum_{i=0}^{n}\binom{n}{i}(1-\frac{1}{m})^{n-i}(\frac{1}{m})^i M_{i+j}\}.$$
$$= 1+k(1-\frac{1}{Q})^{k-1}\sum_{j\geq 0} M_{1+j}p(j)+\delta_B k(1-\frac{1}{Q})^{k-1}(m-1)\sum_{j\geq 0} M_j p(j)$$
$$+Qm\sum_{j\geq 0}p(j)\{\sum_{n=0}^{k}\binom{k}{n}(1-\frac{1}{Q})^{k-n}(\frac{1}{Q})^n\sum_{i=0}^{n}\binom{n}{i}(1-\frac{1}{m})^{n-i}(\frac{1}{m})^i M_{i+j}\}$$
$$-\{Qm(1-\frac{1}{Q})^k+(m-1)k(1-\frac{1}{Q})^{k-1}\}\sum_{j\geq 0} M_j p(j)$$
$$-k(1-\frac{1}{Q})^{k-1}\sum_{j\geq 0} M_{1+j}p(j). \tag{4A.1}$$

Let us define the following:

$$F=\sum_{n=0}^{k}\binom{k}{n}(1-\frac{1}{Q})^{k-n}(\frac{1}{Q})^n\sum_{i=0}^{n}\binom{n}{i}(1-\frac{1}{m})^{n-i}(\frac{1}{m})^i M_{i+j}.$$

Since that $\binom{k}{n}\binom{n}{i}=\binom{k-i}{n-i}\binom{k}{i}$, F becomes

$$F=\sum_{n=0}^{k}\sum_{i=0}^{n}\binom{k-i}{n-i}\binom{k}{i}(1-\frac{1}{Q})^{k-n}(1-\frac{1}{m})^{n-i}(\frac{1}{Qm})^i(\frac{1}{Q})^{n-i}M_{i+j}$$

$$=\sum_{i=0}^{k}\binom{k}{i}M_{i+j}\sum_{h=0}^{k-i}\binom{k-i}{h}(1-\frac{1}{Q})^{k-i-h}\{(1-\frac{1}{m})\frac{1}{Q}\}^h(\frac{1}{Qm})^i \quad (h=n-i)$$

$$=\sum_{i=0}^{k}\binom{k}{i}M_{i+j}(\frac{1}{Qm})^i(1-\frac{1}{Qm})^{k-i}. \tag{4A.2}$$

Finally, substituting Eq.(4A.2) into Eq.(4A.1), we obtain Eq.(4.4):

$$M_k = 1+\sum_{j\geq 0}\sum_{i=0}^{k}\binom{k}{i}(\frac{1}{Qm})^{i-1}(1-\frac{1}{Qm})^{k-i}M_{i+j}p(j)$$
$$-\{Qm(1-\frac{1}{Q})^k+(1-\delta_B)(m-1)k(1-\frac{1}{Q})^{k-1}\}\sum_{j\geq 0}M_j p(j). \quad (k\geq 2)$$

Appendix 4-B Derivation of Eq.(4.8)

By multiplying both sides of Eq.(4.4) by $z^k/k!$ and taking sum of both sides over $k\geq 0$, we have

$$\sum_{k\geq 0} M_k \frac{z^k}{k!} = \sum_{k\geq 0} \frac{z^k}{k!} + \sum_{k\geq 2} \sum_{i=0}^{k} \sum_{j\geq 0} \binom{k}{i} a^{i-1}(1-a)^{k-i} \frac{z^k}{k!} M_{i+j} p(j)$$

$$- \sum_{k\geq 2} \{a^{-1}(1-\frac{1}{Q})^k + (1-\delta_B)(m-1)k(1-\frac{1}{Q})^{k-1}\} \frac{z^k}{k!} \sum_{j\geq 0} M_j p(j). \quad (4B.1)$$

The second term of the right hand side in Eq.(4B.1) becomes

$$\sum_{k\geq 2} \sum_{i=0}^{k} \sum_{j\geq 0} \binom{k}{i} a^{i-1}(1-a)^{k-i} \frac{z^k}{k!} M_{i+j} p(j)$$

$$= \sum_{k\geq 0} \sum_{i=0}^{k} \sum_{j\geq 0} \binom{k}{i} a^{i-1}(1-a)^{k-i} \frac{z^k}{k!} M_{i+j} p(j)$$

$$- \sum_{j\geq 0} \{a^{-1}(1-a)M_j + M_{1+j}\} z p(j) - \sum_{j\geq 0} a^{-1} M_j p(j).$$

Let F denote the first term in the right hand side of the above equation. Then, we have

$$F = a^{-1} \sum_{k\geq 0} \sum_{i=0}^{k} \sum_{j\geq 0} \frac{1}{i!(k-i)!} (az)^i \{(1-a)z\}^{k-i} M_{i+j} \frac{\lambda^j}{j!} e^{-\lambda}$$

$$= a^{-1} \sum_{k\geq 0} \sum_{i=0}^{k} \sum_{j\geq 0} \binom{i+j}{i} (az)^i \lambda^j \frac{\{(1-a)z\}^{k-i}}{(k-i)!} M_{i+j} \frac{1}{(i+j)!} e^{-\lambda}$$

$$= a^{-1} \sum_{n\geq 0} \sum_{i=0}^{n} \binom{n}{i} (az)^i \lambda^{n-i} \sum_{h\geq 0} \frac{\{(1-a)z\}^h}{h!} M_n \frac{1}{n!} e^{-\lambda}$$

$$= a^{-1} \sum_{n\geq 0} e^{z-(\lambda+az)} M_n \frac{(\lambda+az)^n}{n!}.$$

Thus, from the definition of $M^*(z) = \sum_{k\geq 0} M_k \frac{z^k}{k!} e^{-z}$, F becomes

$$F=a^{-1}e^{z}M^{*}(\lambda+az).$$

Therefore, we can rewrite Eq.(4B.1) as

$$\begin{aligned}M(z)=&e^{z}+a^{-1}e^{z}M^{*}(\lambda+az)\\&-a^{-1}\{e^{z(1-1/Q)}-z(1-\frac{1}{Q})-1\}M^{*}(\lambda)\\&-a^{-1}(1-\delta_{B})\{(1-\frac{1}{m})\frac{z}{Q}e^{z(1-1/Q)}-(1-\frac{1}{m})\frac{z}{Q}\}M^{*}(\lambda)\\&-a^{-1}(1-a)zM^{*}(\lambda)-zM^{(1)}(\lambda)e^{-\lambda}-a^{-1}M^{*}(\lambda).\end{aligned}$$

Multiplying the above equation by e^{-z}, we have, from the definition of $M^{*(1)}(\lambda)=M^{(1)}(\lambda)e^{-\lambda}-M^{*}(\lambda)$,

$$M^{*}(z)-a^{-1}M^{*}(\lambda+az)=\begin{cases}1+a^{-1}M^{*}(\lambda)[-\frac{z}{Q}e^{-z}-e^{-z/Q}]+M^{*(1)}(\lambda)(-ze^{-z}). & \text{(TA/M-BF(m))}\\ 1+a^{-1}M^{*}(\lambda)[-aze^{-z}-\{(1-\frac{1}{m})\frac{z}{Q}+1\}e^{-z/Q}] & \\ \quad +M^{*(1)}(\lambda)(-ze^{-z}). & \text{(TA/M-TF(m))}\end{cases}$$

Thus, defining the functions f(z) and g(z) as follows:

$$f(z)=\begin{cases}-\frac{z}{Q}e^{-z}-e^{-z/Q} & \text{(TA/M-BF(m))}\\ -aze^{-z}-\{(1-\frac{1}{m})\frac{z}{Q}+1\}e^{-z/Q}, & \text{(TA/M-TF(m))}\end{cases}$$

$$g(z)=-ze^{-z},$$

we obtain Eq.(4.8):

$$M^{*}(z)-a^{-1}M^{*}(\lambda+az)=1+a^{-1}M^{*}(\lambda)f(z)+M^{*(1)}(\lambda)g(z).$$

Appendix 4-C Derivation of Eq.(4.10)

For a function $\sum_{i\geq 0} a^{-i}\psi^{(2)}(\sigma_M^{[i]}(z))$, we define the following functions:

$$\Theta^{(1)}(\psi(.);z)=\int_0^z \sum_{i\geq 0} a^{-i}\psi^{(2)}(\sigma_M^{[i]}(z))dz,$$

$$\Theta(\psi(.);z)=\int_0^z \Theta^{(1)}(\psi(.);z)dz.$$

Then, as shown in [MATH 85], these functions are given by

$$\Theta^{(1)}(\psi(.);z)=\sum_{i\geq 0}\{\psi^{(1)}(\sigma_M^{[i]}(z))-\psi^{(1)}(\sigma_M^{[i]}(0))\},$$

$$\Theta(\psi(.);z)=\sum_{i\geq 0} a^{-i}\{\psi(\sigma_M^{[i]}(z))-\psi(\sigma_M^{[i]}(0))-a^i z\,\psi^{(1)}(\sigma_M^{[i]}(0))\}.$$

Note that the function $\sigma_M^{[i]}(z)$ satisfies the following three equations:

$$\sigma_M^{[0]}(z)=z,$$

$$\sigma_M^{[i]}(\lambda)=\lambda\frac{1-a^i}{1-a}+\lambda a^i$$

$$=\sigma_M^{[i+1]}(0),$$

$$\lim_{i\to\infty}\sigma_M^{[i]}(z)=\frac{\lambda}{1-a}.$$

Defining

$$\mu=\frac{\lambda}{1-a},$$

we have

$$\Theta^{(1)}(\psi(.);\lambda)=\sum_{i\geq 0}[\psi^{(1)}(\sigma_M^{[i+1]}(0))-\psi^{(1)}(\sigma_M^{[i]}(0))]$$

$$=\psi^{(1)}(\mu)-\psi^{(1)}(0).$$

Thus, the integral of Eq.(4.9) is expressed in terms of $\Theta^{(1)}(\psi(.);z)$ as follows (i.e., Eq.(4.10)):

$$M^{*(1)}(z)=a^{-1}M^{*}(\lambda)\,\Theta^{(1)}(f(.);z)+M^{*(1)}(\lambda)\,\Theta^{(1)}(g(.);z).$$

Appendix 4-D Derivation of Eq.(4.18)

Let Q go to infinity in Eq.(4.4) ($\delta_B=0$ and $a=(Qm)^{-1}$):

$$M_k= 1+ \sum_{j\geq 0} \sum_{i=0}^{k} \binom{k}{i} a^{i-1}(1-a)^{k-i} M_{i+j}p(j)$$

$$-\{Qm(1-\frac{1}{Q})^k+k(m-1)(1-\frac{1}{Q})^{k-1}\} \sum_{j\geq 0} M_j p(j). \tag{4D.1}$$

First, we rewrite Eq.(4D.1) to

$$M_k= 1+ \sum_{j\geq 0} \sum_{i=2}^{k} \binom{k}{i} a^{i-1}(1-a)^{k-i} M_{i+j}p(j)$$

$$+ \sum_{j\geq 0} \binom{k}{0} a^{-1}(1-a)^k M_j p(j)$$

$$+ \sum_{j\geq 0} \binom{k}{1} a^0 (1-a)^{k-1} M_{1+j} p(j)$$

$$-\{Qm(1-\frac{1}{Q})^k+k(m-1)(1-\frac{1}{Q})^{k-1}\} \sum_{j\geq 0} M_j p(j). \tag{4D.2}$$

Let Q go to infinity in Eq.(4D.2). Then, the second term approaches zero and the fourth term approaches $k \sum_{j\geq 0} M_{1+j}p(j)$ in the right hand side of Eq.(4D.2). Then, let R denote the sum of the third and fifth terms; R is given by

$$R= \sum_{j\geq 0} M_j p(j)[Qm(1-\frac{1}{Qm})^k-Qm(1-\frac{1}{Q})^k-k(m-1)(1-\frac{1}{Q})^{k-1}].$$

Note that R=0 when m equals 1. For any finite value of m, R approaches 0 as Q goes to infinity as will shown below. The term in the brackets [ ] becomes

$$Qm \sum_{i=0}^{k} \binom{k}{i}(-\frac{1}{Qm})^i-Qm \sum_{i=0}^{k} \binom{k}{i}(-\frac{1}{Q})^i-(m-1)k \sum_{i=0}^{k-1} \binom{k-1}{i}(-\frac{1}{Q})^i$$

$$=Qm(1-\frac{k}{Qm})-Qm(1-\frac{k}{Q})-(m-1)k$$

$$- \sum_{i=2}^{k} \binom{k}{i}(-\frac{1}{Qm})^{i-1}+m \sum_{i=2}^{k} \binom{k}{i}(-\frac{1}{Q})^{i-1}-(m-1)k \sum_{i=1}^{k-1} \binom{k-1}{i}(-\frac{1}{Q})^i$$

$$= \sum_{i=2}^{k} \binom{k}{i}(-\frac{1}{Qm})^{i-1}+m \sum_{i=2}^{k} \binom{k}{i}(-\frac{1}{Q})^{i-1}-(m-1)k \sum_{i=1}^{k-1} \binom{k-1}{i}(-\frac{1}{Q})^i$$

Thus, $\lim_{Q \longrightarrow \infty} R=0$. Therefore, we obtain

$$\lim_{Q \longrightarrow \infty} M_k = 1 + k \sum_{j \geq 0} M_{1+j}\, p(j).$$

Appendix 4-E Derivation of Eq.(4.19)

Equation (4.18) is rewritten as follows:

$$M_k=1+kM^{(1)}(\lambda)e^{-\lambda} \qquad (k\geq 2) \tag{4E.1}$$

$$M_0=M_1=1.$$

By multiplying Eq.(4E.1) by $z^k/k!$ and taking sum of both sides over $k\geq 0$, we have

$$M(z)=e^z+z(e^z-1)\{M^{*(1)}(\lambda)+M^*(\lambda)\}. \tag{4E.2}$$

Multiplying Eq.(4E.2) by e^{-z} yields

$$M^*(z)=1+z(1-e^{-z})\{M^{*(1)}(\lambda)+M^*(\lambda)\}. \tag{4E.3}$$

Substituting $z=\lambda$ into Eq.(4E.3), we have after simple manipulation

$$M^{*(1)}(\lambda)=M^*(\lambda)\{1/(\lambda(1-e^{-\lambda}))-1\}-1/\{\lambda(1-e^{-\lambda})\}. \tag{4E.4}$$

Furthermore, taking the derivative of Eq.(4E.3) and substituting $z=\lambda$ into the resulting equation, we obtain after simple manipulation

$$M^{*(1)}(\lambda)=\{1/((1-\lambda)e^{-\lambda})-1\}M^*(\lambda). \tag{4E.5}$$

Substituting Eq.(4E.5) into Eq.(4E.2), we have

$$M(z)=e^z+M^*(\lambda)/((1-\lambda)e^{-\lambda})\{z(e^z-1)\}. \tag{4E.6}$$

From Eqs.(4E.4) and (4E.5), we have the following explicit expression of $M^*(\lambda)$:

$$M^*(\lambda)=\{(1-\lambda)e^{-\lambda}\}/(e^{-\lambda}-\lambda). \tag{4E.7}$$

Substituting Eq.(4E.7) into Eq.(4E.6), we get

$$M(z)=e^z+1/(e^{-\lambda}-\lambda)\{z(e^z-1)\}.$$

By expanding the exponential function into power series, we have

$$M(z)=\sum_{k\geq 0}\frac{z^k}{k!}+1/(e^{-\lambda}-\lambda)\sum_{k\geq 2}k\frac{z^k}{k!}$$

Noting that $M(z)=\sum_{k\geq 0}M_k\frac{z^k}{k!}$ and equating the coefficients of $z^k/k!$ of both sides of the above equation, we finally obtain

$$M_k=1+k/(e^{-\lambda}-\lambda) \qquad (k\geq 2),$$

$$M_0=M_1=1.$$

Chapter 5

Throughput Analysis of Reservation Protocols with Tree Type Reservation Channel

5.1 Introduction

Contention-based protocols are suitable for the environment where there are a large population of bursty users. On the other hand, controlled-access protocols are appropriate to users who generate long multipacket messages or users with steady input traffic. Among controlled-access protocols, reservation schemes are well known to achieve high throughput performance. In this chapter, we will focus on reservation access protocols, which are in a class of demand-adaptive protocols (see Fig. 1.1).

In reservation access protocols, it is important how to reserve the channel for sending data packets; i.e., this yields another multiple access problem of what protocol is suitable for the system. Reservation systems are divided into two classes according to whether or not a part of the channel is dedicated to reservation channel (or reservation slot). The Reservation-ALOHA protocol was originally proposed by Crowther et al. [CROW 73], and this protocol introduces "frame" consisting of several slots. Users transmit packets in the enabled slot using ALOHA scheme. Whenever a user successfully transmits a packet in a slot, say i^{th} slot in a frame, he can exclusively use the same slot (i.e., the i^{th} slot) in the succeeding frames until he becomes inactive. This protocol does not provide for reservation channel or a user does not explicitly issue a (reservation) request, so that it has been referred to as an implicit reservation scheme. Lam has theoretically analyzed this system [LAM 80].

Roberts proposed the modified version of this protocol in [ROBE 73], which used reservation slots. In Roberts' protocol, referred to as FIFO reservation scheme, a frame consists of L+1 slots and the L+1st slot in a frame is further partitioned into V small slots, which are reservation slots. A user with a packet to be transmitted randomly chooses one of V small slots and sends a (reservation) request packet. If his request packet is transmitted without collision, then a sequence of future time slots is successfully reserved for his data packet transmission. In other words, all the packets for which a reservation has been made join one "queue in the sky," the length of which is known at all times to all the users under a distributed control, and the packets are served on the basis of FIFO discipline. If a collision occurs in a reservation slot, retransmission is done in such a fashion as the ALOHA scheme (see [ROBE 73, LAM 79, RETN 80] for other reservation access protocols). These ALOHA type reservation schemes achieve very high throughput. However, they suffer from instability phenomenon as well as (direct access) ALOHA scheme [TASA 84, SZPA 83].

Lee and Mark proposed two combined random/reservation access protocols in [LEE 83]. Their intent was to devise high throughput and low delay protocol which took advantage of both random and reservation access. They employed ALOHA scheme and tree algorithm as reservation access protocol, and referred to each protocol as uncontrolled channel access (UCA) and controlled channel access (CCA), respectively. In these protocols, a frame consisted of a data slot and several small slots. Other reservation protocol employing binary tree algorithm has analyzed in [TSYB 80a], which does not provide reservation slots and is, thus, suitable for multipacket messages.

In this chapter, we will analyze two reservation access protocols; one employs a blocked access tree algorithm (BA-RTA) and the other a free access tree algorithm (FA-RTA). In our system, a frame consists of L (large) slots and Q small slots, and thus our protocols are referred to as Q-ary BA-RTA and FA-RTA. The CCA in [LEE 83] is basically equivalent to the Q-ary BA-RTA with L=1. Lee and Mark have obtained that Q=3 offers the optimum performance among the CCA by numerical results for transmission delay. However, the throughput performance of the Q-ary BA-RTA was not explicitly derived. We will explicitly obtain the maximum throughput of the BA-RTA and, furthermore, obtain the maximum throughput of the FA-RTA. Through these analyses, we obtain an optimum frame configuration that maximizes the throughput for a given small-slot length. Our result for the BA-RTA with L=1 agrees with the result for the CCA obtained in [LEE 83].

In section 5.2, an exact description of the above protocols will be given. In sections 5.3.1 and 5.3.2, we will first analyze the maximum throughput of reservation channel in the BA-RTA and the FA-RTA, respectively. In addition, in section 5.3.3, we will obtain an optimum condition of a set {Q, L} that maximizes the throughput for a given length of small slot.

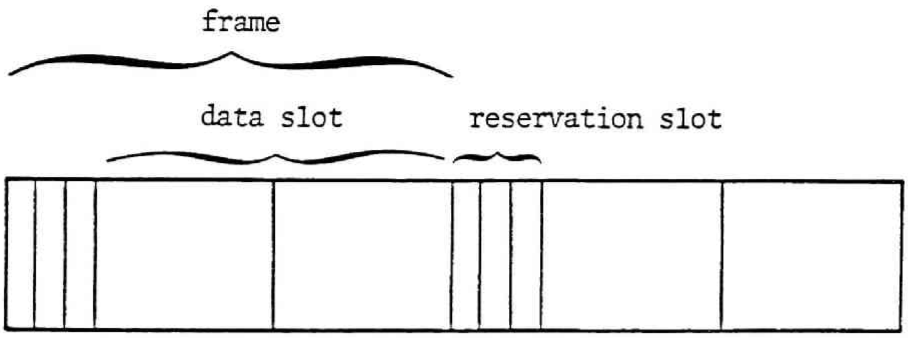


Figure 5.1 Frame configuration (Q=3, L=2)

5.2 Reservation Tree Algorithms

In the following we assume that channel time consists of reservation (small) slots and data slots, and Q small slots and L data slots are organized into a frame (see Fig.5.1). The length of a data slot is equal to data packet transmission time. Furthermore, a message is assumed to be a single packet message.

A user with data packets first sends a request packet in one of Q small slots (in a frame) randomly chosen in order to reserve a data slot. All users can detect the state of a small slot until the next frame starts (i.e., they can distinguish between an empty, a successful and a collided slot). If a request packet is successfully transmitted, a data packet joins the queue for transmission; a data packet is transmitted in the absence of collision. If a collision occurs on the reservation channel, the user retransmits his request packet according to a tree algorithm. We employ a blocked access tree algorithm or a free access tree algorithm to schedule request packet transmission.

Reservation protocol using blocked access tree algorithm (BA-RTA);

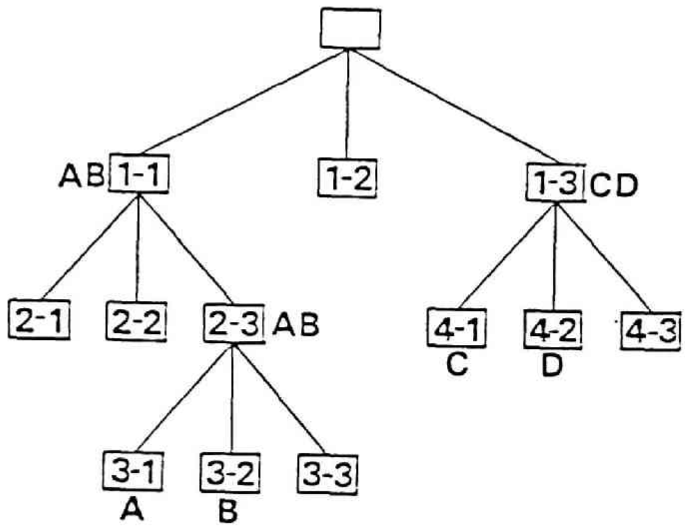
If a frame involves collided small slots, each collided small slot is assigned to a frame. Then, each of the users involved in the same collided small slot chooses one of Q number of small slots in the same frame to retransmit his request packets. This process continues until the initial collision is resolved. If a request packet is transmitted successfully, a data packet is transmitted in a data slot on the basis of FIFO discipline.

Figure 5.2 shows an example of transmission process of a BA-RTA. Users A, B, C and D transmitted their request packets in frame 1. Users A and B chose the first small slot, and users C and D chose the third small slot; both small slots became collided ones. In frame 2, users A and B again choose the same small slot, and at last, in the succeeding frame (frame 3), users A and B choose different small slots, resulting in successful transmissions. Subsequently, users A and B successfully transmit their data packets in data slots in frames 3 and 4, respectively. In a similar way, a collision due to users C and D in frame 1 will be resolved. Since new request packets are forced to wait until the outstanding collision is resolved, users E and F wait until users A, B, C and D successfully transmit their request packets.

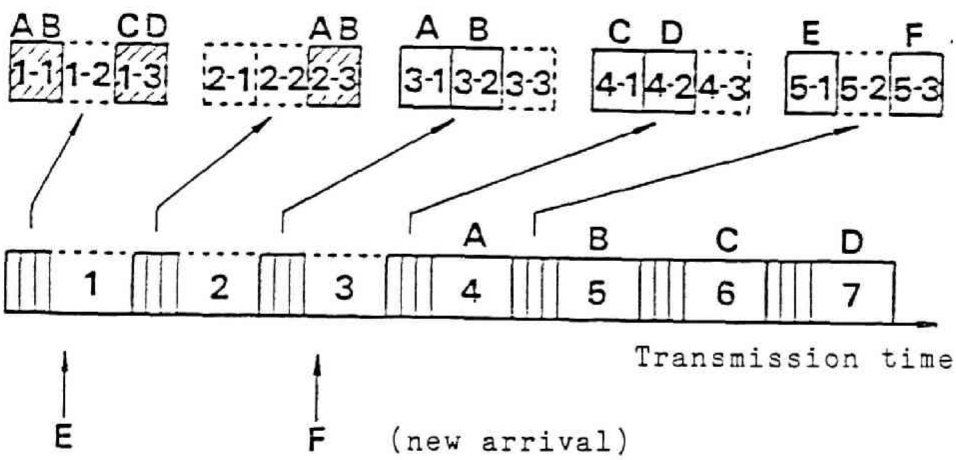
Reservation protocol using free access tree algorithm (FA-RTA);

In this protocol, a new packet is transmitted immediately after its arrival. Others of this protocol are the same as those of the above BA-RTA. Note an FA-RTA is easier to implement than a BA-RTA because an FA-RTA requires each user to monitor the channel only after he becomes active.

Figure 5.3 shows an example of transmission process of an FA-RTA. In contrast to users E and F in Fig. 5.2, users E and F transmit their request packets in frames 2 and 4 immediately after their arrivals, respectively.

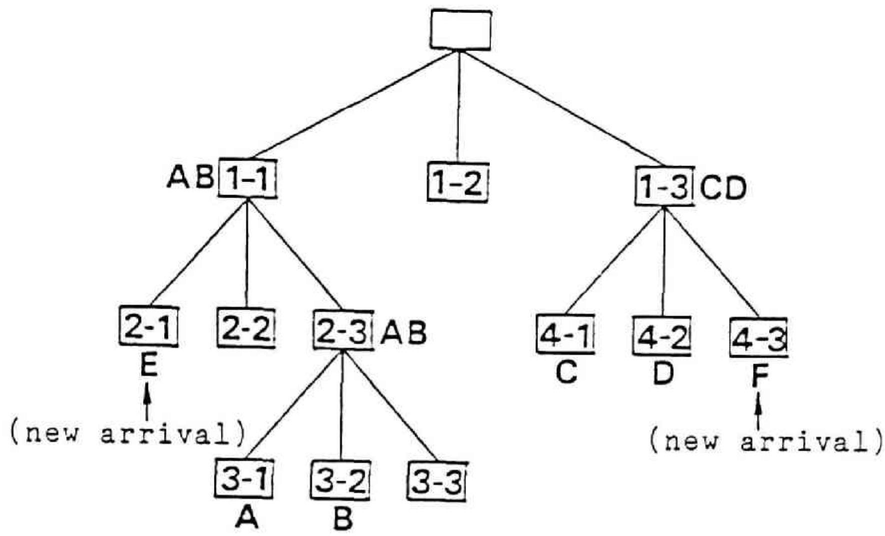


(a) Tree graph for the collision resolution procedure in (b)

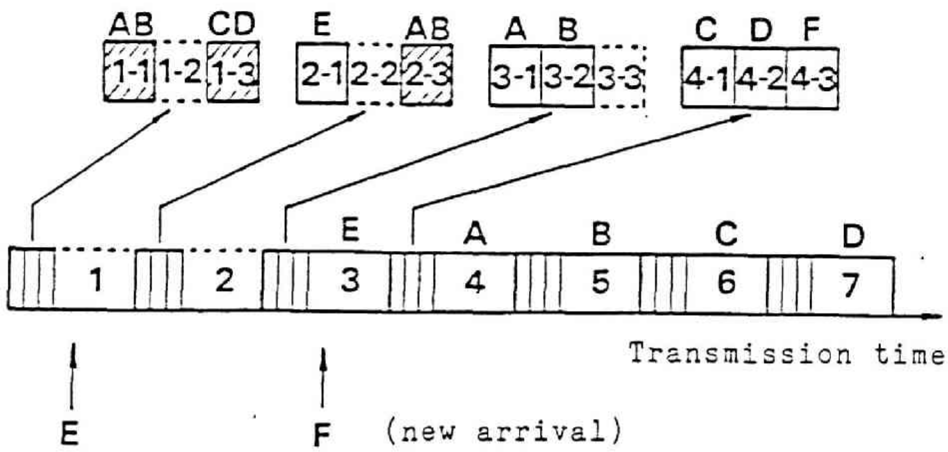


(b) Transmission process

Figure 5.2 Example of transmission process in BA-RTA with Q=3 and L=1



(a) Tree graph for the collision resolution procedure in (b)



(b) Transmission process

Figure 5.3 Example of transmission process in FA-RTA with Q=3 and L=1

5.3. Throughput Analysis of BA-RTA and FA-RTA

In this section, we analyze the maximum throughput of the BA-RTA and the FA-RTA; the maximum throughput of each algorithm is denoted by $\bar{S}_{BA-RTA}(Q,L)$ and $\bar{S}_{FA-RTA}(Q,L)$. In a reservation system with a reservation channel, performance of both reservation channel and data channel determines the maximum throughput of the system. Letting h be the ratio of the length of a reservation slot (small slot) to the length of a data slot, we have

$$\bar{S}_{*-RTA}(Q,L)=\min\left\{\frac{S_{*-RTA}(Q)}{L+hQ},\ \frac{L}{L+hQ}\right\}, \tag{5.1}$$

where $S_{*-RTA}(Q)$ is the maximum throughput under the assumption that L=1 and h=0 in each algorithm. In the following, assuming that new packet arrivals follow a Poisson distribution with parameter λ (packets/frame), we will derive $S_{BA-RTA}(Q)$ (in section 5.3.1) and $S_{FA-RTA}(Q)$ (in section 5.3.2), and further we will obtain an optimum set {Q, L} maximizing the throughput for a given small-slot length (in section 5.3.3).

5.3.1 Analysis of BA-RTA

We refer to the number of request packets transmitted in Q small slots (of a frame) as (collision) multiplicity, and consider the conditional average collision resolution time (CRT) for a given multiplicity. We will obtain the average CRT of a Q-ary BA-RTA in a way similar to that obtaining the average CRT of a blocked access TA/M (in chapter 3). Since both an empty small-slot and a successful small-slot require no frame and only a collided small slot requires a frame to resolve a collision, the conditional probability generating function (pgf) $U_k(z)$ of the CRT for a given multiplicity k satisfies the following equation:

$$U_k(z) = z \sum_{n_1+\dots n_Q=k} p(k,n_1,\dots,n_Q)\left\{\prod_{i=1}^{Q} V_{n_i}(z)\right\}/Q^k, \quad (k\geq 2), \tag{5.2}$$

where

$$V_n(z)=\begin{cases} 0 & (n=0,1) \\ U_n(z) & (n\neq 0,1). \end{cases}$$

If k is equal to 0 or 1, one frame is used, so that

$$U_0(z)=U_1(z)=z.$$

In Eq. (5.2), $p(k,n_1,\ldots,n_Q)$ is the multinomial coefficient defined by

$$p(k,n_1,\ldots,n_Q) = k!/(n_1!\ldots.n_Q!).$$

We denote the average CRT of a BA-RTA by B_k. Then, B_k is given by $\frac{d}{dz}U_k(z)|_{z=1}$. By differentiating (5.2) with respect to z and setting z to 1, the following equation is obtained:

$$B_k = 1+\sum_{n_1+\ldots n_Q=k} p(k,n_1,\ldots,n_Q)[\sum_{i=1}^{Q} v_{n_i}]/Q^k \quad (k\geq 2), \tag{5.3}$$

where v_n is given by $\frac{d}{dz}V_n(z)|_{z=1}$ and thus

$$v_n = \begin{cases} 0 \ (n=0,1) \\ B_n \ (n\geq 2). \end{cases}$$

Substituting v_n into Eq. (5.3), we obtain the following recurrence equation for B_k:

$$B_k = 1-Q(1-q)^k-k(1-q)^{k-1}+Q\sum_{n=2}^{k}\binom{k}{n}(1-q)^{k-n}q^nB_n, \tag{5.4}$$

where q is defined to be 1/Q.

We will solve this equation for B_k. First let us define the following generating function:

$$B(z) = \sum_{k\geq 0} B_kz^k/k!. \tag{5.5}$$

By multiplying both sides of Eq. (5.4) by $z^k/k!$, summing over $k\geq 0$ and using Eq. (5.5), we have

$$B(z)-QB(qz)e^{(1-q)z} = e^{Qz}-(z+Q)e^{(1-q)z}. \tag{5.6}$$

Further, defining $B^*(z)$ by

$$B^*(z) = e^{-z}B(z), \tag{5.7}$$

Eq. (5.6) becomes

$$B^*(z)-QB^*(qz) = 1-(z+Q)e^{-qz}.$$

Here, $B^*(z)$ is represented as $B^*(z)=\sum_{n\geq 0} B^*_n z^n$. Then equating coefficients of z^n in the above equation, B^*_n is given by

$$B^*_n = \begin{cases} 1 \quad (n=0) \qquad 0 \qquad (n=1) \\ (n-1)(-1)^n q^{n-1}/\{(1-q^{n-1})n!\} \qquad (n\geq 2). \end{cases} \tag{5.8}$$

From the definitions (5.5) and (5.7), we have the following equation for B_k and B^*_n:

$$B_k = k! \sum_{n=0}^{k} B^*_n/(k-n)!.$$

Thus, from Eq. (5.8), we finally obtain the following closed-form solution for B_k:

$$B_k = 1+\sum_{n=2}^{k}(-1)^n(n-1)\binom{k}{n}/(Q^{n-1}-1) \qquad (k\geq 2). \tag{5.9}$$

Now, we define $S_{BA-RTA}(Q)$ as follows:

$$S_{BA-RTA}(Q) = \liminf_{k\longrightarrow\infty}\frac{k}{B_k}.$$

Then, if an input rate is less than $S_{BA-RTA}(Q)$, a system is stable. Thus, $S_{BA-RTA}(Q)$ represents a stable maximum throughput (or a supremum of the throughput) if L=1 and h=0. In order to obtain $S_{BA-RTA}(Q)$, we must estimate the asymptotic behavior of k/B_k when multiplicity k approaches infinity.

Here, let T_k be the average CRT of the Q-ary basic tree algorithm. Then, T_k is given by ([MATH 85, MURO 85])

$$T_k = 1+Q+Q\sum_{n=2}^{k}(-1)^n(n-1)\binom{k}{n}/(Q^{n-1}-1) \quad (k\geq 2). \tag{5.10}$$

We define the maximum throughput $S_{BA-TA}(Q)$ in the basic tree algorithm as follows:

$$S_{BA-TA}(Q) = \liminf_{k \longrightarrow \infty} \frac{k}{T_k}.$$

Recently, we have made clear the asymptotic behaviour of the function k/T_k for a given positive integer Q (see section 3.4). It has been shown in section 3.4 that, as the value k approaches infinity, k/T_k does not converge to any function which is independent of value k, but converges to some function which oscillates around ln(Q)/Q. Furthermore, from the practical point of view, we can estimate $S_{BA-TA}(Q)$ as follows (see section 3.4 and Appendix 3-D):

$$S_{BA-TA}(Q) \fallingdotseq \ln(Q)/Q. \tag{5.11}$$

From Eqs.(5.9) and (5.10), the relation between T_k and B_k is given by

$$B_k = T_k/Q-1-1/Q. \tag{5.12}$$

Accordingly, it follows from Eq. (5.12) that $S_{BA-RTA}(Q)$ is written in terms of $S_{BA-TA}(Q)$ as

$$\begin{aligned} S_{BA-RTA}(Q) &= S_{BA-TA}(Q)Q \\ &= \ln(Q). \end{aligned} \tag{5.13}$$

Figure 5.4 plots the throughput k/B_k versus multiplicity k of a Q-ary BA-RTA. In the same figure, the values of $S_{BA-RTA}(Q)$ are plotted for corresponding values of Q. This figure shows that $S_{BA-RTA}(Q)$ is appropriately estimated.

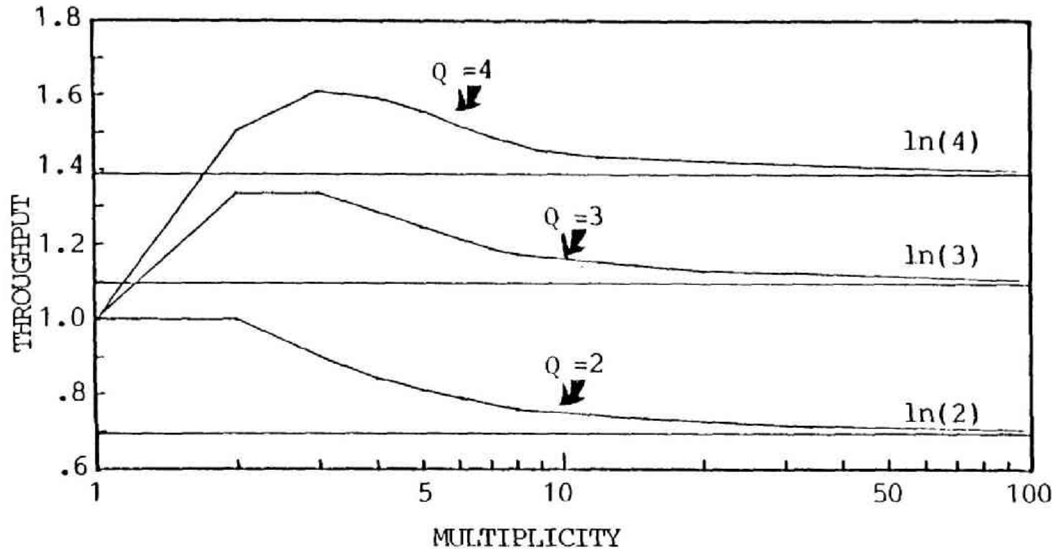


Figure 5.4 Throughput $(\frac{k}{B_k})$ vs. multiplicity and ln(Q) in BA-RTA

5.3.2. Analysis of FA-RTA

As in section 5.3.1, we will obtain the time required to resolve a collision (i.e., CRT) for a given multiplicity k. We should note that, as a result of a free access mechanism, new packets join previously collided packets in each subtree. Let $W_k(z)$ be the conditional pgf of the CRT of a Q-ary FA-RTA for a given multiplicity k. A collided tree is first partitioned into Q subtrees randomly. When a subtree contains at most one request packet, no frame is used, and a frame is only assigned to each collided subtree (i.e., each subtree containing at least two request packets). As a result, $W_k(z)$ becomes

$$W_k(z)=z \sum_{n_1+\ldots+n_Q=k} p(k,n_1,\ldots,n_Q)\{\prod_{i=1}^{Q} V_{n_i}(z)\}/Q^k \quad (k\geq 2), \tag{5.14}$$

where

$$V_i(z)=\begin{cases} 1 & (i=0,1) \\ \sum_{j\geq 0} p(j)W_{n+j}(z) & (i\neq 0,1). \end{cases} \tag{5.15}$$

If k is equal to zero or one, only one frame is used;

$$W_0(z)=W_1(z)=z.$$

In Eq. (5.15), p(j) is the probability that j new packets arrive in a frame;

$$p(j)=e^{-\lambda}\,\lambda^j/j!,$$

and $p(k,n_1,\ldots,n_Q)$ is the multinomial coefficient (see the preceding section). Let us denote the average conditional CRT of a Q-ary FA-RTA by R_k, from Eqs. (5.14) and (5.15), we have

$$R_k=1+Q\sum_{i=2}^{k}\binom{k}{i}(1-q)^{k-i}q^i\sum_{j\geq 0} p(j)R_{i+j}, \tag{5.16}$$

where $q=1/Q$.

We obtained the recurrence equation (5.16) for R_k. Subsequently, we will obtain the closed form solution to this equation. First, we introduce the following functions as in Chapter 3:

$$F(z)=\sum_{n\geq 0} R_n z^n/n!,$$

$$F^*(z)=e^{-z}F(z).$$

Taking the derivative of these functions, we have

$$F^{(1)}(z)=\sum_{n\geq 0} R_{n+1} z^n/n!,$$

$$F^{*(1)}(z)=e^{-z}(F^{(1)}(z)-F(z)).$$

By multiplying both sides of Eq. (5.16) by $z^k/k!$ and taking sum of both sides over $k\geq 0$, we have

$$F^*(z)-q^{-1}F^*(\lambda+qz)=1+q^{-1}F^*(\lambda)f(z)+F^{*(1)}(\lambda)g(z) \tag{5.17}$$

where

$$f(z)=-(1+qz)e^{-qz},$$

$$g(z)=-ze^{-qz}.$$

By differentiating Eq. (5.17) twice with respect to z, we have

$$F^{*(2)}(z)-qF^{*(2)}(\lambda+qz)=q^{-1}F^*(\lambda)f^{(2)}(z)+F^{*(1)}(\lambda)g^{(2)}(z).$$

This equation has the following solution (see [MATH 85]):

$$F^{*(2)}(z)=q^{-1}F^*(\lambda)\sum_{i\geq 0} q^i f^{(2)}(\sigma_M^{[i]}(z))+F^{*(1)}(\lambda)\sum_{i\geq 0} q^i g^{(2)}(\sigma_M^{[i]}(z)), \tag{5.18}$$

where

$$\sigma_M^{[i]}(z)=\lambda(1-q^i)/(1-q)+q^i z.$$

Thus, integrating Eq.(5.18) yields

$$F^{*(1)}(z)=q^{-1}F^*(\lambda)\,\Theta^{(1)}(f(.);z)+F^{*(1)}(\lambda)\,\Theta^{(1)}(g(.);z) \tag{5.19}$$

(see Appendix 4-C for this integration and the definition of

$\Theta^{(1)}(f(.);z))$. Therefore, letting $z=\lambda$ in Eq. (5.19),

$$F^{*(1)}(\lambda)=q^{-1}F^{*}(\lambda)\Theta^{(1)}(f(.);\lambda)/\{1-\Theta^{(1)}(g(.);\lambda)\}.$$
$$=QF^{*}(\lambda)q^{2}\mu/(1-q\mu), \tag{5.20}$$

where $\mu=\lambda/(1-q)$. In order to obtain $F^{*}(z)$, we will integrate Eq. (5.19) with respect to z. This integration leads to the following equation:

$$F^{*}(z)=1+QF^{*}(\lambda)d^{*}(z), \tag{5.21}$$

where

$$d^{*}(z)=\Theta(f(.);z)+q^{2}\mu/(1-q\mu)\Theta(g(.);z).$$

Consequently, by multiplying Eq. (5.21) by e^{z} and equating the coefficients of $z^{k}/k!$ on both sides, we obtain the average conditional CRT, R_k, as follows:

$$R_k=1+QF^{*}(\lambda)d_k,$$

where

$$d_k=1/(1-\nu)e^{-\nu}\,[\nu\sum_{m\geq 0}\exp(\nu q^{m})\{(1-q^{m+1})^{k}+kq^{m+1}-1\}$$
$$+\sum_{m\geq 0}Q^{m}\{1-(1-q^{m+1})^{k}-kq^{m+1}(1-q^{m+1})^{k-1}\}]$$

$$\nu=q\mu.$$

We will now derive the maximum throughput, which will be denoted by $S_{FA-RTA}(Q)$. First, we should note from its definition that $F^{*}(\lambda)$ refers to the average CRT when new packets arrive according to Poisson process with parameter λ. Thus, the stability condition of this algorithm is that $F^{*}(\lambda)$ takes a finite positive value. We have already obtained the important Eq.(5.21) related to this value in course of obtaining R_k. Setting $z=\lambda$ in Eq. (5.20) yields

$$F^{*}(\lambda)=1/(1-Qd^{*}(\lambda)). \tag{5.22}$$

Therefore, we find that the maximum throughput $S_{FA-RTA}(Q)$ is λ which satisfies the following equation:

$$Qd^{*}(\lambda)=1. \tag{5.23}$$

From some manipulation, $d^{*}(\lambda)$ becomes

$$d^{*}(\lambda)= q\lambda/\{1-(\lambda+1)q\}e^{-q\{\lambda/(1-q)\}}$$
$$\cdot\sum_{j\geq 1}\{q\lambda/(1-q)\}^{j}\frac{j}{(j+1)!}\{j(1-q)/(1-q^{j})-q^{j}\}. \tag{5.24}$$

Here, in the free access basic tree algorithm, we denote the average CRT by $L^{*}(\lambda)$, which corresponds to $F^{*}(\lambda)$ in an FA-RTA. The equation corresponding to Eq. (5.22) is written as

$$L^{*}(\lambda)=1/(1-Qd^{*}_{TA}(\lambda)), \tag{5.24}$$

where $d^{*}_{TA}(\lambda)$ was obtained in [MATH 85] as follows:

$$d^{*}_{TA}(\lambda)= \lambda/\{1-\lambda-q\}e^{-\lambda/(1-q)}$$
$$\cdot\sum_{j\geq 1}\{\lambda/(1-q)\}^{j}\frac{j}{(j+1)!}\{j(1-q)/(1-q^{j})-q\}. \tag{5.25}$$

From Eqs. (5.24) and (5.25), it holds that

$$d^{*}(\lambda)=d^{*}_{TA}(q\lambda).$$

Thus, it is clear that the maximum throughput of the Q-ary FA-RTA is as Q times as that of the Q-ary free access basic TA.

5.3.3 Optimum Frame Configuration

In the preceding sections, we obtained the throughput $S_{BA-RTA}(Q)$ and $S_{FA-RTA}(Q)$. So far we have neglected the length of a small slot, denoted by h, and a frame configuration. In this section, we consider the maximum throughput in the system where a frame consists of L (large) slots and Q number of small slots. Recall Eq. (5.1) related to the maximum throughput in such a system:

$$\bar{S}_{*-RTA}(Q,L)=\min\ \{\frac{S_{*-RTA}(Q)}{L+hQ},\ \frac{L}{L+hQ}\}.$$

We will obtain an optimum set {Q, L} (i.e., $\{Q_{opt}, L_{opt}\}$) which maximizes $\bar{S}_{*-RTA}(Q,L)$, where * is used instead of BA or FA. It follows from Eq. (5.1) that for a fixed value of Q, $L=S_{*-RTA}(Q)$ gives $\bar{S}_{*-RTA}(Q,L)$ the maximum value, i.e.,

$$1/\{1+hQ/S_{*-RTA}(Q)\}.$$

Since, in a real system, Q and L take integer values,

$$L_{opt}=\lfloor S_{*-RTA}(Q)\rfloor \text{ or } \lfloor S_{*-RTA}(Q)\rfloor+1.$$

In the BA-RTA or the FA-RTA, a set of $\{L_{opt}, Q_{opt}\}$ is as follows.

(a) BA-RTA

From Eq. (5.12), $1/\{1+hQ/S_{BA-RTA}(Q)\}$ achieves the highest value when Q equals e. Since Q and L take integer values, an optimum set is that $Q_{opt}=3$ and $L_{opt}=1$ for any value of h such that $0<h<1$; thsi optimum set provides the maximum throughput

$$\bar{S}_{BA-RTA}(3,1) = 1/(1+3h).$$

We give several values of $S_{BA-RTA}(Q)$ for $2\leq Q\leq 5$ in Table 5.1, and values of $\bar{S}_{BA-RTA}(Q,1)$ for various values of h in Table 5.2.

(b) FA-RTA

In the FA-RTA, we can obtain the value of $S_{FA-RTA}(Q)$ by solving Eq.(5.23). Table 5.1 shows several values of $S_{FA-RTA}(Q)$ for $2\leq Q\leq 5$. In Table 5.3, we give the value of L_{opt} and the corresponding value of

$\bar{S}_{FA-RTA}(Q,L_{opt})$ for $2 \leq Q \leq 5$. From this table, we have the following results:

(i) $h<0.0789$; an optimum set $\{Q_{opt}, L_{opt}\}$ such that Q_{opt} and L_{opt} satisfy both $Q_{opt}/L_{opt}=3$ and $S_{FA-RTA}(Q_{opt}) > L$, e.g., $\{3, 1\}$, $\{6, 2\}$, $\{9, 3\}$, leads to the maximum throughput

$\bar{S}_{FA-RTA}(Q_{opt},L_{opt})=1/(1+3h)$,

(ii) $h>0.0789$; $L_{opt}=2$ and $Q_{opt}=5$ leads to the maximum throughput

$\bar{S}_{FA-RTA}(5,2)=S_{FA-RTA}(5)/(2+5h)$.

Table 5.1

$S_{BA-RTA}(Q)$ and $S_{FA-RTA}(Q)$

Q	S_{BA-RTA}	S_{FA-RTA}
2	0.6931	0.7203
3	1.0986	1.2048
4	1.3863	1.5969
5	1.6094	1.9362

Table 5.2

Optimum values of L
and
corresponding values of $\bar{\bar{S}}_{BA-RTA}(Q,L_{opt})$

h	S(2,L) (L)	S(3,L) (L)	S(4,L) (L)	S(5,L) (L)
0.010	0.67956(1)	0.97087(1)	0.96154(1)	0.95238(1)
0.015	0.67296(1)	0.95694(1)	0.94340(1)	0.93023(1)
0.020	0.66649(1)	0.94340(1)	0.92593(1)	0.90909(1)
0.025	0.66014(1)	0.93023(1)	0.90909(1)	0.88889(1)
0.030	0.65931(1)	0.91743(1)	0.89286(1)	0.86957(1)
0.035	0.64780(1)	0.90498(1)	0.87719(1)	0.85106(1)
0.040	0.64180(1)	0.89286(1)	0.86207(1)	0.83333(1)
0.045	0.63591(1)	0.88106(1)	0.84746(1)	0.81633(1)
0.050	0.63013(1)	0.86957(1)	0.83333(1)	0.80000(1)
0.055	0.62446(1)	0.85837(1)	0.81967(1)	0.78431(1)
0.060	0.61888(1)	0.84746(1)	0.80645(1)	0.76923(1)
0.065	0.61340(1)	0.83682(1)	0.79365(1)	0.75472(1)
0.070	0.60802(1)	0.82645(1)	0.78125(1)	0.74074(1)
0.075	0.60274(1)	0.81633(1)	0.76923(1)	0.72727(1)
0.080	0.59754(1)	0.80645(1)	0.75758(1)	0.71429(1)
0.085	0.59243(1)	0.79681(1)	0.74627(1)	0.70175(1)
0.090	0.58741(1)	0.78740(1)	0.73529(1)	0.68966(1)
0.095	0.58248(1)	0.77821(1)	0.72464(1)	0.67797(1)
0.100	0.57762(1)	0.76923(1)	0.71429(1)	0.66667(1)

Table 5.3

Optimum values of L
and
corresponding values of $\tilde{S}_{FA-RTA}(Q,L_{opt})$

h	S(2,L) (L)	S(3,L) (L)	S(4,L) (L)	S(5,L) (L)
0.010	0.70623(1)	0.97087(1)	0.96154(1)	0.95238(1)
0.015	0.69937(1)	0.95694(1)	0.94340(1)	0.93311(2)
0.020	0.69265(1)	0.94340(1)	0.92593(1)	0.92200(2)
0.025	0.68605(1)	0.93023(1)	0.90909(1)	0.91116(2)
0.030	0.67958(1)	0.91743(1)	0.89286(1)	0.90056(2)
0.035	0.67323(1)	0.90498(1)	0.87719(1)	0.89021(2)
0.040	0.66699(1)	0.89286(1)	0.86207(1)	0.88009(2)
0.045	0.66088(1)	0.88106(1)	0.84746(1)	0.87021(2)
0.050	0.65487(1)	0.86957(1)	0.83333(1)	0.86054(2)
0.055	0.64897(1)	0.85837(1)	0.81967(1)	0.85108(2)
0.060	0.64317(1)	0.84746(1)	0.80645(1)	0.84183(2)
0.065	0.63748(1)	0.83682(1)	0.79365(1)	0.83278(2)
0.070	0.63189(1)	0.82645(1)	0.78125(1)	0.82395(2)
0.075	0.62639(1)	0.81633(1)	0.76923(1)	0.81525(2)
0.080	0.62099(1)	0.80645(1)	0.75758(1)	0.80675(2)
0.085	0.61569(1)	0.79681(1)	0.74627(1)	0.79844(2)
0.090	0.61047(1)	0.78740(1)	0.73529(1)	0.79029(2)
0.095	0.60534(1)	0.77821(1)	0.72464(1)	0.78231(2)
0.100	0.60030(1)	0.76923(1)	0.71429(1)	0.77448(2)

5.4 Concluding Remarks

In this chapter, we have considered two reservation protocols (the BA-RTA and the FA-RTA); as a scheduling algorithm of an access to reservation channel, one exploits the blocked access tree algorithm, and the other does the free access tree algorithm. We analytically derived the maximum throughput of both algorithms. Through our analyses, an optimum frame configuration was obtained for a given small-slot length, h, as follows:

(1) in the BA-RTA, for any value of h such that $0<h<1$,

a set of $\{Q_{opt}=3, L_{opt}=1\}$ provides the maximum throughput:

$\bar{S}_{BA-RTA}(3,1) = 1/(1+3h)$, and

(2) in the FA-RTA, if $h<0.0789$,

a set of $\{3, 1\}$ (or $\{6, 2\}$, $\{9, 3\}$) provides the maximum throughput:

$\bar{S}_{FA-RTA}(Q_{opt},L_{opt})=1/(1+3h)$;

otherwise (i.e., if $h>0.0789$),

a set of $\{5, 2\}$ provides the maximum throughput

$\bar{S}_{FA-RTA}(5,2)=S_{FA-RTA}(5)/(2+5h)$.

Thus, if $h<0.0789$, $\bar{S}_{BA-RTA}(Q_{opt},L_{opt}) = \bar{S}_{FA-RTA}(Q_{opt},L_{opt})$; otherwise, $\bar{S}_{BA-RTA}(Q_{opt},L_{opt}) < \bar{S}_{FA-RTA}(Q_{opt},L_{opt})$.

For the BA-RTA with L=1, Lee and Mark [LEE 83] obtained $Q_{opt}=3$ through average transmission delay analysis. Our result agrees with their result. It is noteworthy that the FA-RTA requires users to monitor the channel only when they have packets to send in contrast to the BA-RTA. Therefore, comparing the BA-RTA and the FA-RTA, we have seen that the FA-RTA is superior to the BA-RTA in both maximum throughput and the ease of implementation.

Chapter 6

Concluding Remarks

6.1 Summary of this thesis

In this thesis, we have considered a subset of tree-based collision resolution algorithms (CRAs) in contention-based communication systems. Contention-based protocols offer an effective way to allocate the channel among a large number of bursty users. However, several authors had pointed out the underlying issue of instability (or saturation) phenomenon in those protocols, and then proposed the solutions employing adaptive controls in response to the changing quantities such as transmission rate and the number of backlogged users. Tsybakov [TSYB 78] and Capetanakis [CAPE 79a,b] attacked this problem and devised a novel approach to overcome this instability phenomenon without the control mechanism adaptive to dynamically changing quantities above stated but by means of a tree structure in scheduling transmission. Owing to such high innovation of contention-based protocols, extensive works on tree-based CRAs have done after the works of Tsybakov and Capetanakis. The outline of the studies on the topics and the positions of our studies in the class of tree-based CRAs were shown in Fig. 1.4. The main purpose of this thesis was to develop several new branches of tree-based CRAs taking care of practical applications.

First, we would like to just review this thesis based on the chapter organization.

In chapter 2, we proposed a blocked access tree algorithm with Q number of (control) mini-slots (Q-ary BA TA/M), referred to as the adaptive tree algorithm, to improve Capetanakis' tree algorithm. This algorithm is an address-based partitioning algorithm (see section

1.2.2.2), and we assumed that each user distinguished between an empty mini-slot and a busy mini-slot: i.e., binary feedback information. We approximately analyzed the average transmission delay for the algorithm by taking a propagation delay into account. The analysis agreed with simulation results over the wide range of an input rate; yet, at heavy traffic, the results of simulation experiments also suggested that there was room for further improvement in approximation. Our analysis showed that the adaptive tree algorithm performs fairly efficient compared to the basic tree algorithm, and that the performance of the adaptive tree algorithm becomes better as Q becomes larger.

In chapter 3, we considered two blocked access Q-ary TA/Ms (BA TA/M-BF and BA TA/M-TF) based on a probabilistic partitioning algorithm. One assumed binary feedback information on the state of mini-slot was available (this was also treated in chapter 2), and the other assumed ternary feedback information; each user further distinguished between a mini-slot involving only one signal and a mini-slot involving at least two signals in a busy mini-slot. We treated a BA TA/M-TF where m slots were assigned to a collided mini-slot (i.e., the users having sent signals in the collided mini-slot) for retransmissions, so that we referred to the BA TA/M-TF as the BA TA/M-TF(m). Note that the BA TA/M-TF(1) is equivalent to the BA TA/M-BF. We presented a way to evaluate the stable maximum throughput as a function of Q for the above two BA TA/Ms. Our way will be easily applied to other BA tree type algorithm if its average CRT is obtained in the closed form. Though numerical computations, it was seen that, if a length of mini-slot (h) is small (i.e., $h<0.017$), the BA TA/M-BF is superior to the BA TA/M-TF(m) ($m\geq 2$). The BA TA/M-BF is also excellent in that it needs only binary something/nothing feedback information in mini-slot.

In chapter 4, we studied two free access TA/Ms (FA TA/Ms): the FA TA/M-BF(m) and the FA TA/M-TF(m). The maximum throughput and the upper bound of throughput (=0.56714) were analytically obtained. Through the analysis, for a given mini-slot length, we obtained an optimum number of mini-slots maximizing throughput. Furthermore, we explicitly obtained the lower bound of the average transmission delay. The lower bound was also the lower bound of the delay in the whole class of free access algorithms including both algorithms with and without mini-slots. It was shown that the FA TA/M-BF(1) is the best among all of the FA TA/M-BF(m)s ($m \geq 1$), and the FA TA/M-TF(2) is the best among all of the FA TA/M-TF(m)s ($m \geq 1$). Since the FA TA/M-BF(1) is equivalent to the FA TA/M-TF(1), the FA TA/M-TF(2) is the best among the FA TA/Ms considered in chapter 4. In addition, from numerical and simulation results, we concluded that an FA TA/M-BF and an FA TA/M-TF provided a practical way to achieve performance close to the best performance of all the free access algorithms.

In chapter 5, we studied two reservation systems (BA RTA and FA RTA); a BA RTA exploited a blocked access tree-based CRA, and an FA RTA did a free access tree-based CRA to schedule an access to the reservation channel. It was assumed that Q small-slots and L data sub-slots were organized into a frame. By analyzing throughput characteristic of both systems, we obtained an optimum frame configuration maximizing throughput for a given small-slot length. For the reasons of the ease of implementation as well as high throughput, it has become clear that a free access reservation system is superior to the other.

Here, we will summarize the performance of the above stated tree algorithms. Figure 6.1 shows a comparison of the maximum throughput of

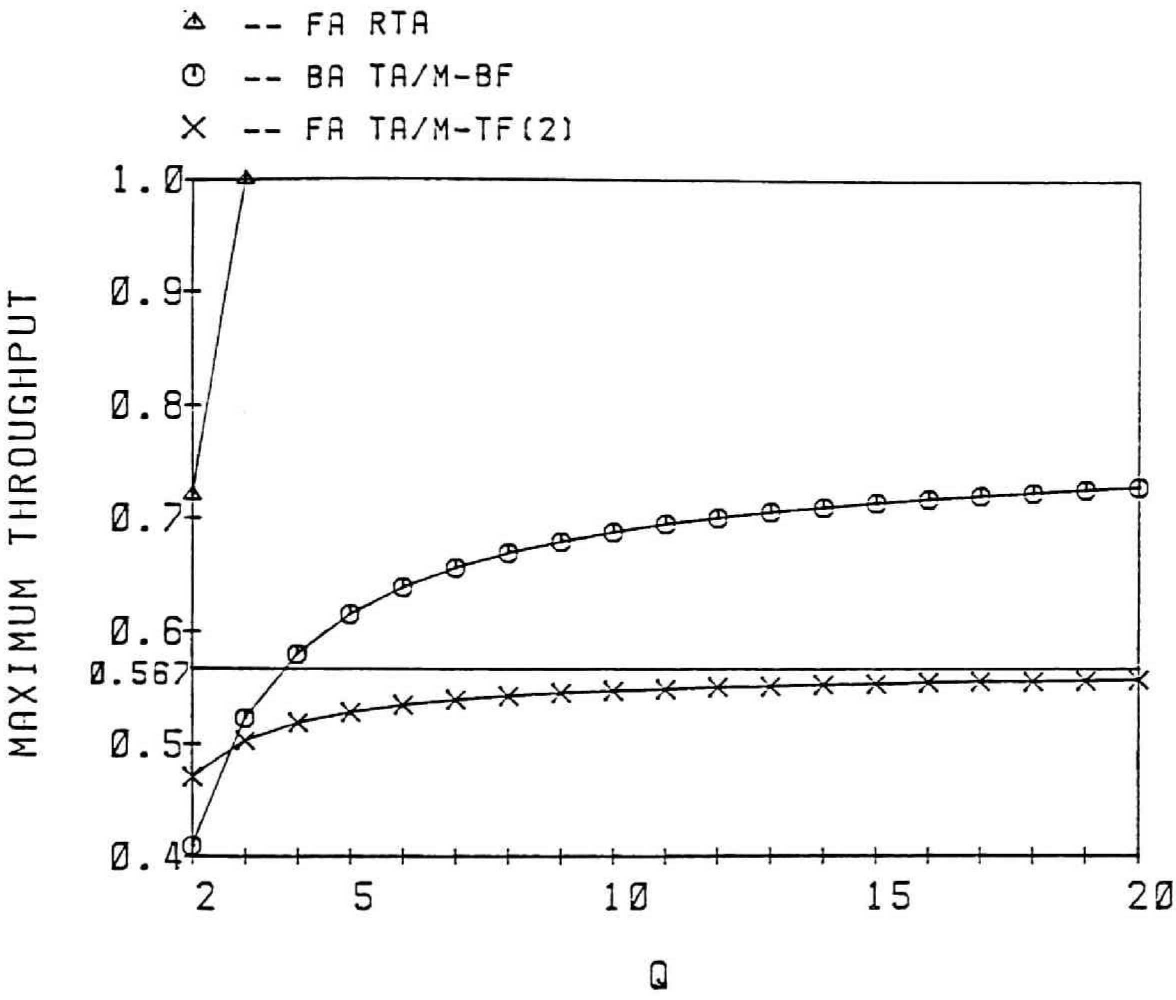


Figure 6.1 Maximum throughput of several algorithms analyzed in this thesis

several algorithms treated in this thesis as a function of Q. In this figure, a length of mini-slot and that of (reservation) small-slot are assumed to be zero. In addition to this assumption, in an FA RTA, a frame is assumed to consist of Q small-slots and one data sub-slot.

Consider a collision resolution interval (CRI), which represents an interval from an initial collision to the instant at which all the packets involved in its collision are successfully transmitted. As previously mentioned, in a BA TA/M-BF, empty slots are completely eliminated in a CRI with the aid of the information provided by mini-slots. New packets are forced to wait for an end of a CRI (on the basis of a blocked access mechanism). These result in good performance as shown in Fig. 6.1. However, a BA requires each user to monitor even when he has no packets; i.e., it requires continuous sensing.

In an FA TA/M-BF, a collision resolution mechanism is the same as that of a BA TA/M, but new packets are transmitted immediately after their arrivals without regard for a collision resolution procedure of the outstanding packets (on the basis of a free access mechanism). For this reason, new packets transmitted in a CRI always interfere with a collision resolution of the outstanding packets. Therefore, we can expect the performance degradation due to a free access mechanism. Nevertheless, we were interested in a limited sensing property of a free access mechanism; a property that its mechanism requires each user to monitor the channel only when he has a packet to be transmitted. Since the maximum achievable throughput is 0.56714 in the context of free access as stated in chapter 4, Fig. 6.1 shows that an FA TA/M achieves maximum throughput close to the upper bound of all the free access algorithms with values of Q such that $Q \geq 20$.

In addition, for an FA RTA, Figure 6.1 shows that only 3 small-slots provide ideal throughput 1.0 if an overhead (i.e., a length of small-slot) is neglected.

Consequently, from Fig. 6.1, we can observe a natural result that more complex mechanism provides more efficient performance. In particular, we see that a blocked access mechanism is very useful in a class of TA/Ms. The results of these analyses are useful in choosing an appropriate algorithm which meets given conditions: e.g., required (throughput vs. delay) performance and permissible complexity of implementation.

6.2 Issues for future research

We have considered some issues on CRAs in this dissertation. As previously mentioned, CRAs have already been extensively and broadly studied. However, we can still address the issues of interest to be attacked in this field as follows.

(1) The performance evaluation of tree-based algorithms in an environment which suffers from a long propagation delay; chapter 2 treated this problem, and an exact or a well approximate analysis is desirable. As stated in [CAPE 79a], there are two ways to execute the tree search: i.e., serial and parallel tree searches. The comparison of these two searches is also interesting in such a system. Furthermore, Tsybakov [TSYB 81] discussed an effect of propagation delay on performance.

(2) The study on limited sensing tree algorithms; the FA TA/Ms analyzed in chapter 4 are also one of limited sensing algorithms. This class of algorithms is of practical interest because of the ease of implementation, and has thus been treated in several papers such as [GEOR 85, HUMB 86]. However, the limited sensing algorithms have not

been so improved in throughput. It is practically important to devise a more efficient limited sensing algorithm and analyze its performance.

(3) The performance analysis of address-based tree algorithms; an address-based tree algorithm offers the desirable property that a collision is neccessarily resolved with a finite number of retransmissions; for example, if the total number of users in a system is equal to Q^d, the number of retransmissions experienced by a packet is less than or equal to d in a Q-ary tree algorithm. This property may particularly help to improve the performance of a system with a small population size (see [SHIM 86]).

REFERENCES

Abbreviations

ICC (International Conference on Communications)
IECEJ (The Institute of Electronics and Communication Engineering of Japan)
IPS (Information Processing Society of Japan)

[ABRA 70] Abramson, N., "The ALOHA System - Another Alternative for Computer Communications," Proc. of FJCC, pp.281-285, 1970.

[BERG 81] Berger, T., "The Poisson Multiple-Access Conflict Resolution Problem," in Multi-User Communication Systems, G. Longo ed., CISM Courses and Lectures, No.265, pp.1-27, Springer Verlag, 1981.

[BERG 83] Berger, T. and Huang, J.C., "A Conflict Resolution Algorithm with Multibit Feedback," Technical Report, School of Engineering, Cornell University, Ithaca, NY. submitted to IEEE Trans. Infor. Theory , Dec. 1983.

[BERG 84] Berger, T., Mehravari, N., Towsley, D. and Wolf, J.K., "Random Multiple-Access Communication and Group Testing," IEEE Trans. Commun., Vol.COM-32, No.7, pp.769-778, 1984.

[BUX 81] Bux, W., "Local-Area Subnetworks: A Performance Comparison," IEEE Trans. Commun., Vol.COM-29, No.10, pp.1465-1473, 1981.

[CAPE 79a] Capetanakis, J.I., "Tree Algorithm for Packet Broadcast Channels," IEEE Trans. Infor. Theory, Vol.IT-25, No.5, pp.505-515, 1979.

[CAPE 79b] Capetanakis, J.I., "Generalized TDMA: The Multi-Accessing Tree Protocol," IEEE Trans. Commun., Vol.COM-27, No.10, pp.1476-1484, 1979.

[CARL 75] Carleial, A.B. and Hellman, M.E., "Bistable Behavior of ALOHA Type Systems," IEEE Trans. Commun., Vol.COM-23, No.4, pp.401-410, 1975.

[CROW 73] Crowther, W.R. et al., "A System for Broadcast Communication: Reservation-ALOHA," Proc. 6th Hawaii Int. Conf. Syst. Sci., 1973.

[CRUZ 82] Cruz, R. and Hajek, B., "A New Upper-Bound to the Throughput of a Multiple-Access Broadcast Channel," IEEE Trans. Infor. Theory, Vol.IT-28, No.3, pp.402-405, 1982.

[FAYO 77] Fayolle, G. et al., "Stability and Optimal Control of the Packet Switching Broadcast Channel," J. ACM, Vol.24, pp.375-386, 1977.

[FAYO 85] Fayolle, G., Flajolet, P., Hofri, M. and Jacquet, P., "Analysis of a Stack Algorithm for Random Multiple-Access Communication," IEEE Trans. Infor. Theory, Vol.IT-31, No.2, pp.244-254, 1985.

[FERG 75] Ferguson, M., "On the Control, Stability and Waiting Time in a Slotted ALOHA Random Access System," IEEE Trans. Commun., Vol.COM-23, No.11, pp.1306-1311, 1975.

[GALL 78] Gallager, R.G., "Conflict Resolution in Random Access Broadcast Networks," Proc. the AFSOR Workshop in Commun. Theory and Applications, pp.74-76, Sep. 1978

[GALL 85] Gallager, R.G., "A Perspective on Multiaccess Channels," IEEE Trans. Infor. Theory, Vol.IT-31, No.2, pp.124-142, 1985

[GEOR 83] Georgiadis, L. and Papantoni-Kazakos, P., "A Collision Resolution Protocol for Random Access Channels with Energy Detectors," IEEE Trans. Commun., Vol.COM-30, No.11, pp.2413-2420, 1983.

[GEOR 85] Georgiadis, L. and Papantoni-Kazakos, P., "Limited Feedback Sensing Algorithms for the Packet Broadcast Channel," IEEE Trans. Infor. Theory, Vol.IT-31, No.2, pp.280-294, 1985.

[GRAM 80] Grami, A. and Hayes, J.F., "Delay Performance of Adaptive Local Distribution," Proc. IEEE ICC'80, pp.39.4.1-39.4.5, 1980.

[GRAM 82] Grami, A., Sohraby, K. and Hayes, J., "Further Results on Probing," Proc. IEEE ICC'82, pp.1C.3.1-1C.3.3, 1982.

[GUDJ 80] Gudjohnsen, E.P., Towsley, D. and Wolf, J.K., "On Adaptive Polling Techniques for Computer Communication Networks," Proc. IEEE ICC'80, pp.13.3.1-13.3.5, 1980.

[HAYE 78] Hayes, J.F., "An Adaptive Technique for Local Distribution," IEEE Trans. Commun., Vol.COM-26, No.8, pp.1178-1186, 1978.

[HAYE 81] Hayes, J.F., "An Adaptive Technique for Local Distribution," IEEE Commun. Magazine, Vol.19, No.2, pp.6-14, 1981.

[HENR 77] Henrici, P., "Applied and Computational Complex Analysis, Vol.2," John Wiley & Sons, New York , 1977.

[HOFR 84] Hofri, M., "Stack Algorithms for Collision-Detecting Channels and Their Analysis: A Limited Survey," in Lecture Notes in Control and Information Sciences 60 (Proc. of the inter. seminar on modelling and performance evaluation methodology, INRIA, 1983), Baccelli, F. and Fayolle, G. eds., Springer Verlag, Berlin, pp.71-88, 1984.

[HUAN 83] Huang, J.C. and Xu, B.Z., "Pretested Random Tree Protocols for Packet Conflict Resolution," IEEE Trans. Infor. Theory, Vol. IT-29, No.6, pp.910-914, 1983.

[HUAN 85] Huang, J.C., "Interval-Searching Algorithms for Multiple-Access Contention Resolution: Delay and Throughput," Ph.D Thesis, School of Electrical Eng., Cornell Univ., January 1985.

[HUMB 86] Humblet, P.A., "On the Throughput of Channel Access Algorithms with Limited Sensing," IEEE Trans. Commun., Vol.COM-34, No.4, pp.345-347, 1986.

[IEEE 85] IEEE Trans. Infor. Theory (edited by Gallager, R.G.) Special Issues on Random-Access Communications Vol.IT-31, No.2, 1985.

[JACO 78] Jacobs, I.M., Binder, R. and Hoverstein, E.V., "General Purpose Satellite Networks," Proc. IEEE, Vol.66, No.11, pp.1448-1467, 1978.

[KLEI 75a] Kleinrock, L. and Lam, S.S., "Packet Switching in a Multi-access Broadcast Channel: Performance Evaluation," IEEE Trans. Commun., Vol.COM-23, No.4, pp.410-423, 1975.

[KLEI 75b] Kleinrock, L., "Queueing Systems Vol.1: Theory," John Wiley & Sons, 1975.

[KLEI 76] Kleinrock, L., "Queueing Systems Vol.2: Computer Applications," John Wiley & Sons, 1976.

[KLEI 78] Kleinrock, L. and Yemini, Y., "An Optimal Adaptive Scheme for Multiple Access Broadcast Communication," Proc. IEEE ICC'78, pp.7.2.1-7.2.5, 1978.

[KLEI 80] Kleinrock, L. and Scholl, M.O., "Packet Switching in Radio Channels: New Conflict-Free Multiple Access Schemes," IEEE Trans. Commun., Vol.COM-28, No.7, pp.1015-1029, 1980.

[KLEI 85] Kleinrock, L., "On Queueing Problems in Random-Access Communications," IEEE Trans. Infor. Theory, Vol.IT-31, No.2, pp.166-175, 1985.

[KURO 84] Kurose, J., Schwartz, M and Yemini, Y., "Multiple-Access Protocols and Time-Constrained Communication," ACM Computing Surveys, Vol.16, No.1, pp.43-70, March 1984.

[LAM 75] Lam, S.S. and Kleinrock, L., "Packet Switching in a Multi-access Broadcast Channel: Dynamic Control Procedures," IEEE Trans. Commun., Vol.COM-23, No.9, pp.891-904, 1975.

[LAM 79] Lam, S.S., "Satellite Packet Communication-Multiple Access Protocols and Performance," IEEE Trans. Commun., Vol.COM-27, No.10, pp.1456-1466, 1979.

[LEE 83] Lee, H.W. and Mark, J.W., "Combined Random/Reservation Access for Packet Switched Transmission over a Satellite with On-board Processing: Part I - Global Beam Satellite," IEEE Trans. Commun., Vol.COM-31, No.10, pp.1161-1171, 1983.

[MASS 80] Massey, J.L., "Collision-Resolution Algorithms and Random-Access Communications," Tech. Rep. UCLA-ENG-8016, School of Eng. and Applied Science, Univ. of California, Los Angeles, April 1980.

[MATH 84] Mathys, P., "Analysis of Random-Access Algorithm," Ph.D Thesis, ETH No.7713, ETH Zurich, Nov. 1984.

[MATH 85] Mathys, P. and Flajolet, P., "Q-ary Collision Resolution Algorithms in Random-Access Systems with Free or Blocked Channel Access," IEEE Trans. Infor. Theory, Vol.IT-31, No.2, pp.217-243, 1985.

[MEDI 83] Meditch, J.S. and Lea, C.A., "Stability and Optimization of the CSMA and CSMA/CD Channels," IEEE Trans. Commun., Vol.COM-31, No.6, pp.763-774, 1983.

[MEHR 84] Mehravari, N. and Berger, T., "Poisson Multiple-Access Contention with Binary Feedback," IEEE Trans. Infor. Theory, Vol.IT-30, No.5, pp.745-751, 1984.

[MEHR 84] Mehravari, N., "TDMA in a Random-Access Environment:an Overview," IEEE Commun. Magazine, Vol.22, No.11, pp.54-59, 1984.

[MERA 83] Merakos, L. and Kazakos, D., "Multiaccess of a Slotted Channel Using a Control Mini-Slot," Proc. IEEE ICC'83, pp.C5.3.1-C5.3.6, 1983.

[MERA 84] Merakos, L. and Bisdikian, C., "Delay Analysis of the n-Ary Stack Algorithm for a Random-Access Broadcast Channel," Proc. 22nd Allerton Conf. on Comm. Contr. Comp., pp.385-394, Oct. 1984.

[MERA 85] Merakos, L., Bisdikian, C. and Exley, G.M., "Throughput-Delay Characteristics of a Stack Algorithm for Local Area Networks using CSMA and CSMA/CD Channels," Proc. IEEE ICC'85, pp.3.2.1-3.2.5, 1985.

[METC 73] Metcalfe, R.M., "Steady-state Analysis of a Slotted and Controlled ALOHA System with Blocking," Proc. 6th Hawaii Int. Conf. Syst. Sci., Jan. 1973.

[METC 76] Metcalfe, R.M. and Boggs, D.R., "Ethernet: Distributed Packet Switching for Local Computer Networks," Commun. ACM, Vol.19, pp.395-404, 1976.

[MIKH 81] Mikhailov, V.A. and Tsybakov, B.S., "Upperbound for the Capacity of Random Multiple Access System," Prob. Inform. Transmission, Vol.17, No.1, pp.63-67, 1981; translated from Problemy Peredachi Informatsii, Vol.17, No.1, pp.90-95, 1981.

[MITT 81] Mittal, K.K. and Venetsanopoulous, A.N., "On the Dynamic Control of the URN Scheme for Multiple Access Broadcast Communication Systems," IEEE Trans. Commun., Vol.COM-29, No.7, pp.962-970, 1981.

[MOLL 81] Molle, M.L., "Unified and Extentions of the Multiple Access Communications Problem," Ph.D Thesis, UCLA-ENG-8118, Computer Science Department, University of California, July 1981.

[MOLL 82] Molle, M.L., "On the Capacity of Infinite Population Multiple Access Protocols," IEEE Trans. Infor. Theory, Vol.IT-28, No.3, pp.396-401, 1982.

[MOSE 85] Mosely, J. and Humblet, P.A., " A Class of Efficient Contention Resolution Algorithms for Multiple Access Channels," IEEE Trans. Commun., Vol.COM-33, No.2, pp.145-151, 1985.

[MURO 85] Muro, S., Oie, Y. and Hasegawa, T., "A Proof of Channel Stability of D-ary Symmetrical Tree Algorithms," Proc. Inter. Conf. on Modelling Techniques and Tools for Performance Analysis, AFCET, France, pp.349-364, 1985 (also in Modelling Techniques and Tools for Performance Analysis '85, N. Abu El Ata ed., North-Holland, pp.343-358, 1986).

[MURO 86a] Muro, S., Oie, Y. and Hasegawa, T., "Tree Type Collision Resolution Algorithms in Multiple Access Communication Protocols," J. IPS Japan, Vol.27, No.5, pp.508-522, 1986 (in Japanese).

[MURO 86b] Muro, S., Oie, Y., Kawano, H. and Hasegawa, T., "Tree Algorithms for Random Access Channels in Local Area Networks," Proc. IEEE ICC'86, pp.19.4.1-19.4.5, 1986.

[OIE 84] Oie, Y., Muro, S. and Hasegawa, T., "Analysis of Adaptive Tree Protocol in a Satellite Communication," Trans. IECEJ, Vol.J67-D, No.6, pp.685-692, 1984 (in Japanese).

[OIE 85a] Oie, Y., Muro, S. and Hasegawa, T., "Throughput Analysis of Tree Type Protocols" Trans. IECEJ, Vol.J68-D, No.6, pp.1226-1233, 1985 (in Japanese).

[OIE 85b] Oie, Y., Muro, S., and Hasegawa, T., "Tree Algorithm for Reservation Multiple Access," Proc. IEEE ICC'85, pp.3.5.1-3.5.5, 1985.

[OIE 86] Oie, Y., Muro, S. and Hasegawa, T., "Throughput Analysis of Reservation Protocols with Tree Type Reservation Channel," Trans. IECEJ, Vol.J69-D, No.4, pp.602-612, 1986 (in Japanese).

[OIE 87] Oie, Y., Suda, T., Miyahara, H. and Hasegawa, T., "Throughput and Delay Evaluations of Free Access Tree Algorithm with Mini-Slots," to appear in Proc. the IEEE INFOCOM'87, 1987.

[PAKE 69] Pakes, A.G., "Some Conditions for Ergodicity and Recurrence of Markov Chains," Oper. Res., Vol.17, pp.1058-1061, 1969.

[PANW 85] Panwar, S.S., Towsely, D. and Wolf, J.K., "On the Throughput of Degenerate Intersection and First-Come First-Served Collision Resolution Algorithms," IEEE Trans. Infor. Theory, Vol. IT-31, No.2, pp.274-279, 1985.

[PIPP 81] Pippenger, N., "Bounds on the Performance of Protocols for a Multiple Access Broadcast Channel," IEEE Trans. Infor. Theory, Vol.IT-27, No.2, pp.145-151, 1981.

[RETN 80] Retnadha, C., "Satellite Multiple Access Protocols," IEEE Commun. Magazine, Vol.18, No.5, pp.16-20, 1980.

[ROBE 73] Roberts, L.G., "Dynamic Allocation of Satellite Capacity through Packet Reservation," Proc. AFIPS Conf., Vol.42, pp.711-716, 1973.

[ROSE 84] Rosenkrantz, W.A., "Some Theorems on the Instability of the Exponential Back-Off Protocol," Performance'84 edited by E. Gelenbe, North-Holland, pp.199-205, 1984.

[RUGE 81] Ruget, G., "Some Tools for the Study of Channel-Sharing Algorithms," in Multi-User Communication Systems, G. Longo ed., CISM Courses and Lectures, No.265, pp.201-237, Springer Verlag, 1981.

[SCHW 77] Schwartz, M., "Computer-Communication Network Design and Analysis," Prentice Hall, 1977.

[SESH 85] Seshadri, N. and Srikantakumar, P.R., "On Group Testing Protocols for Binary-Feedback Multiaccess Channel," IEEE Trans. Commun., Vol.COM-33, No.6, pp.574-577, 1985.

[SHIM 86] Shima, N., Muro, S., Oie, Y., and Hasegawa, T., "Performance Evaluation of a Deterministic Tree Collision Resolution Algorithm," to appear in Proc. International Computer Symposium (ICS'86), 1986.

[SOBE 59] Sobel, M. and Groll, P.A., "Group Testing to Eliminate Efficiency All Defectives in a Binomial Sample," Bell Syst. Tech. J., Vol.38, pp.1179-1252, 1959.

[SZPA 83] Szpankovski, W., "Analysis and Stability Consideration in a Reservation Multiaccess System," IEEE Trans. Commun., Vol.COM-31, No.5, pp.684-692, 1983.

[SZPA 85] Szpankowski, W., "Analysis of a Recurrence Equation Arising in Stack-type Algorithms for Collision-Detecting Channels," Proc. Inter. Seminar on Computer Networking and Performance Evaluation, pp.9.3.1-9.3.12, 1985.

[TANE 81] Tanenbaum, A.S., "Computer Networks," Prentice-Hall, Inc., 1981.

[TASA 84] Tasaka, S. and Ishibashi, Y., "A Reservation Protocol for Satellite Packet Communication - A Performance Analysis and Stability Consideration," IEEE Trans. Commun., Vol.COM-32, No.8, pp.920-927, 1984.

[THOM 84] Thomopoulos, S.C.A., "0.51968: A New Upper Bound on the Efficiency of any Collision-Resolution Algorithm for Random Access

Broadcast Channel," Proc. of 1984 Conf. on Infor. Sci. & Sys., pp.135-139, 1984.

[TOBA 75] Tobagi, F.A. and Kleinrock, L., "Packet-Switching in Radio Channels: Part I-Carrier Sense Multiple Access Modes and Their Throughput-Delay Characteristics," IEEE Trans. Commun., Vol.COM-23, No.2, pp.1400-1416, 1975.

[TOBA 76] Tobagi, F.A. and Kleinrock, L., "Packet-Switching in Radio Channels: Part III-Polling and (Dynamic) Split Channel Reservation Multiple Access," IEEE Trans. Commun., Vol.COM-24, No.8, pp.832-845, 1976.

[TOBA 77] Tobagi, F.A. and Kleinrock, L., "Packet-Switching in Radio Channels: Part IV-Stability Considerations and Dynamic Control in Carrier Sense Multiple Access," IEEE Trans. Commun., Vol.COM-25, No.10, pp.1103-1120, 1977.

[TOBA 80] Tobagi, F.A., "Multiaccess Protocols in Packet Communication Systems," IEEE Trans. Commun., Vol.COM-28, No.4, pp.468-488, 1980.

[TOWS 82] Towsley, D. and Venkatesh, G., "Window Random Access Protocols for Local Computer Networks," IEEE Trans. Comput., Vol.C-31, No.8, pp.715-722, 1982.

[TOWS 84] Towsley, D. and Wolf, J.K., "On Adaptive Tree Polling Algorithms," IEEE Trans. Commun., Vol.COM-32, No.12, pp.1294-1298, 1984.

[TROP 83] Tropper, C., "On Retransmission Protocols for Ethernet," Proc. IEEE ICC'83, pp.C5.7.1-C5.7.4, 1983.

[TSYB 78] Tsybakov, B.S. and Mikhailov, V.A., "Free Synchronous Packet Access in a Broadcast Channel with Feedback," Prob. Inform. Transmission, Vol.14, No.4, pp.259-280, 1978; translated from Problemy Peredachi Informatsii, Vol.14, No.4, pp.32-59, 1978.

[TSYB 80a] Tsybakov, B.S. and Berkovskii, M.A., "Multiple Access with Reservation," Prob. Inform. Transmission, Vol.16, No.1, pp.35-54, 1980; translated from Problemy Peredachi Infomatsii, Vol.16, No.1, pp.50-76, 1980.

[TSYB 80b] Tsybakov, B.S., "Resolution of a Conflict of Known Multiplicity," Prob. Inform. Transmission, Vol.16, No.2, pp.134-144, 1980; translated from Problemy Peredachi Informatsii, Vol.16, No.2, pp.69-82, 1980.

[TSYB 80c] Tsybakov, B.S. and Vvedenskaya, N.D., "Random Multiple-Access Stack Algorithm," Prob. Inform. Transmission, Vol.16, No.3, pp.230-243, 1980; translated from Problemy Peredachi Infomatsii, Vol.16, No.3, pp.80-946, 1980.

[TSYB 80d] Tsybakov, B.S. and Mikhailov, V.A., "Random Multiple Packet Access: Part-and-Try Algorithm," Prob. Inform. Transmission, Vol.16, No.4, pp.305-317, 1980; translated from Problemy Peredachi Informatsii, Vol.16, No.4, pp.65-79, 1980.

[TSYB 81] Tsybakov, B.S., Mikhailov, V.A. and Fedortsev, S.P., "Allowance for Packet Propagation Time in Random Multiple Access," Prob. Inform. Transmission, Vol.17, No.2, pp.131-134, 1981; translated from Problemy Peredachi Informatsii, Vol.17, No.2, pp.75-78, 1981.

[TSYB 85] Tsybakov, B.S., "Survey of USSR Contributions to Random Multiple-Access Communications," IEEE Trans. Infor. Theory, Vol.IT-31, No.2, pp.143-165, 1985.

[WOLF 85] Wolf, J.K., "Born Again Group Testing: Multiaccess Communications," IEEE Trans. Infor. Theory, Vol.IT-31, No.2, pp.185-191, 1985.